

# On the Effectiveness of Sampling for Evolutionary Optimization in Noisy Environments<sup>\*</sup>

Chao Qian<sup>1</sup>, Yang Yu<sup>1</sup>, Yaochu Jin<sup>2</sup>, and Zhi-Hua Zhou<sup>1</sup>

<sup>1</sup>National Key Laboratory for Novel Software Technology,  
Nanjing University, Nanjing 210023, China

<sup>2</sup>Department of Computing, University of Surrey, Guildford, Surrey, GU2 7XH, UK  
{qianc, yuy, zhouzh}@lamda.nju.edu.cn, yaochu.jin@surrey.ac.uk

**Abstract** Sampling has been often employed by evolutionary algorithms to cope with noise when solving noisy real-world optimization problems. It can improve the estimation accuracy by averaging over a number of samples, while also increasing the computation cost. Many studies focused on designing efficient sampling methods, and conflicting empirical results have been reported. In this paper, we investigate the effectiveness of sampling in terms of rigorous running time, and find that sampling can be ineffective. We provide a general sufficient condition under which sampling is useless (i.e., sampling increases the running time for finding an optimal solution), and apply it to analyzing the running time performance of (1+1)-EA for optimizing OneMax and Trap problems in the presence of additive Gaussian noise. Our theoretical analysis indicates that sampling in the above examples is not helpful, which is further confirmed by empirical simulation results.

## 1 Introduction

Evolutionary algorithms (EAs) [4] inspired from natural phenomena are often applied to solve real-world optimization problems, where the fitness (i.e., objective) evaluation of a solution is usually noisy. For example, in airplane design, the fitness of every prototype is evaluated by a stochastic computer simulation, and thus is a random variable whose value can be different from the exact fitness. Handling noise in fitness evaluations is important in that a poor solution can appear to be good due to the noise, which can mislead the search direction, resulting in an inefficient optimization. Many studies thus have focused on dealing with noise in evolutionary optimization [2,6,18].

One simple and direct way to reduce the effect of noise is sampling, which samples the fitness of one solution several times and then uses the average to estimate the true fitness. An  $n$ -time random sampling can reduce the standard deviation by a factor of  $\sqrt{n}$ , and thus makes the fitness estimation closer to the true value, while also increasing the computation cost  $n$  times. Much effort has been devoted to designing smarter sampling approaches, which dynamically decide the sample size for each solution so that the sampling cost is reduced as much as possible.

---

<sup>\*</sup> This research was supported by the National Science Foundation of China (61375061, 61333014) and the Jiangsu Science Foundation (BK2012303).

Aizawa and Wah [1] suggested two adaptive sampling methods: increasing the sample size with the generation number and allocating larger sample size for solutions with larger estimated variance. Stagge [24] used a larger sample size for better solutions. Several sequential sampling approaches [7,8,10] were later proposed for tournament selection, which first estimate the fitness of two solutions by a small number of samples, and then sequentially increase samples until the difference can be significantly discriminated. Adaptive sampling was then incorporated into diverse metaheuristic algorithms (e.g., immune algorithm [27], particle swarm optimization [5] and compact differential evolution [17]) to efficiently cope with noise. It has also been employed by evolutionary algorithms for noisy multi-objective optimization [20,23,25]. Based on the assumption that the fitness landscape is locally smooth, an alternative approach to approximately increase the estimation accuracy without increasing the sampling cost was proposed [9,22], which estimates the fitness of a solution by averaging the fitness of previously evaluated neighbors.

Sampling has been shown to be able to improve the local performance of EAs (e.g., increase the probability of selecting the true better solution in tournament selection [7]). A practical performance measure of an algorithm is how much time it needs to find a desired solution. On this measure, conflicting conclusions about sampling have been empirically reported. For example, in [1], it was shown that sampling can speed up a standard genetic algorithm on two test functions; while in [10], sampling led to a larger computation time for a simple generational genetic algorithm on the OneMax function.

In this paper, we investigate the effectiveness of sampling via rigorous running time analysis, which measures how soon an algorithm can solve a problem (i.e., the number of fitness evaluations until finding an optimal solution) and has been a leading theoretical aspect for randomized search heuristics [3,19]. We provide a sufficient condition under which sampling is useless (i.e., sampling increases the running time). Applying it to analyze (1+1)-EA solving the Noisy OneMax and the Noisy Trap problems with the additive Gaussian noise, we disclose that the sampling is ineffective in the two cases for different reasons. The derived theoretical results are also empirically verified. The results may help understand the effect of noise and design better strategies for handling noisy fitness functions.

The rest of this paper is organized as follows. Section 2 introduces some preliminaries. Section 3 presents the main theorem, which is then used in case studies in Section 4. Section 5 concludes the paper and discusses future work.

## 2 Preliminaries

### 2.1 Sampling and Optimization in the Presence of Noise

An optimization problem can be generally represented as  $\arg \max_{x \in \mathcal{X}} f(x)$ , where  $\mathcal{X}$  is the feasible solution space and the objective  $f$  is also called fitness in the context of evolutionary computation. In real-world optimization tasks, the fitness evaluation for a solution is usually disturbed by noise due to a wide range of uncertainties (e.g., randomized simulations), and consequently we can not obtain the exact fitness value but only a noisy one. A commonly studied noise model is additive noise as presented in Definition 1, which will also be adopted in this paper.

**Definition 1 (Additive Noise).** Given a distribution  $\mathcal{N}$ , let  $f^{\mathcal{N}}(x)$  and  $f(x)$  denote the noisy and true fitness of a solution  $x$  respectively, then

$$f^{\mathcal{N}}(x) = f(x) + \delta,$$

where  $\delta$  is randomly drawn from  $\mathcal{N}$ , denoted by  $\delta \sim \mathcal{N}$ .

In evolutionary optimization, sampling as described in Definition 2 has often been used to reduce the effect of noise. It approximates the true fitness  $f(x)$  by the average of a number of random samples.

**Definition 2 (Sampling).** Sampling of size  $k$  outputs the fitness of a solution as

$$f_k^{\mathcal{N}}(x) = \frac{1}{k} \sum_{i=1}^k (f(x) + \delta_i), \text{ where } \delta_i \sim \mathcal{N}.$$

For additive Gaussian noise (i.e.,  $\mathcal{N} = N(\theta, \sigma^2)$ ),  $f_k^{\mathcal{N}}(x)$  actually can be represented by  $f(x) + \delta$  with  $\delta \sim N(\theta, \sigma^2/k)$ , that is, sampling of size  $k$  reduces the variance of noise by a factor of  $k$  and thus estimates the fitness more accurately. However, the computation time is also increased by  $k$  times. Many studies thus focused on designing efficient sampling methods [1,8,24], while the effectiveness of sampling, in particular a theoretical understanding of sampling, remains unclear.

## 2.2 Evolutionary Algorithms by Markov Chain Analysis

Evolutionary algorithms [4] are a kind of randomized metaheuristic optimization algorithms. Starting from an initial set of solutions (called a population), EAs try to iteratively improve the population by a cycle of three stages: reproducing new solutions from the current population, evaluating the newly generated solutions, and updating the population by removing bad solutions. The (1+1)-EA, as shown in Algorithm 1, is a simple EA for maximizing pseudo-Boolean problems over  $\{0, 1\}^n$ , which reflects the common structure of EAs. It maintains only one solution, and repeatedly tries to improve the current solution by using bit-wise mutation (i.e., step 3) and selection (i.e., steps 4-5). It has been widely used for the running time analysis of EAs, e.g., in [16,26]. For (1+1)-EA with sampling in noisy environments, step 4 changes to be “if  $f_k^{\mathcal{N}}(x') \geq f_k^{\mathcal{N}}(x)$ ”.

**Algorithm 1 ((1+1)-EA)** Given pseudo-Boolean function  $f$  with solution length  $n$ , it consists of the following steps:

1.  $x :=$  randomly selected from  $\{0, 1\}^n$ .
2. Repeat until the termination condition is met
3.  $x' :=$  flip each bit of  $x$  independently with probability  $p$ .
4. if  $f(x') \geq f(x)$
5.  $x := x'$ .

where  $p \in (0, 0.5)$  is the mutation probability.

The evolution process goes forward only based on the current population, thus, an EA can be modeled and analyzed as a Markov chain  $\{\xi_t\}_{t=0}^{+\infty}$  (e.g., in [16,26]) by taking the EA’s population space  $\mathcal{X}$  as the chain’s state space, i.e.  $\xi_t \in \mathcal{X}$ . Let  $\mathcal{X}^* \subset \mathcal{X}$  denote

the set of all optimal populations, which contains at least one optimal solution. The goal of the EA is to reach  $\mathcal{X}^*$  from an initial population. Thus,  $\mathcal{X}^*$  is the optimal state space of the corresponding Markov chain. In this paper, we assume that the Markov chain is homogeneous, since EAs often employ time-invariant operators.

Given a Markov chain  $\{\xi_t\}_{t=0}^{+\infty}$  and  $\xi_{\hat{t}} = x$ , we define its *first hitting time* (FHT) as a random variable  $\tau$  such that  $\tau = \min\{t | \xi_{\hat{t}+t} \in \mathcal{X}^*, t \geq 0\}$ . That is,  $\tau$  is the number of steps needed to reach the optimal space for the first time starting from  $\xi_{\hat{t}} = x$ . The mathematical expectation of  $\tau$ ,  $\mathbb{E}[\tau | \xi_{\hat{t}} = x] = \sum_{i=0}^{\infty} iP(\tau = i)$ , is called the *expected first hitting time* (EFHT). For the corresponding EA, the running time is usually defined as the number of fitness evaluations until an optimal solution is found for the first time, since the fitness evaluation is often the computational process with the highest cost [16,26]. Thus, the *expected running time* of the EA starting from  $\xi_0$  is equal to  $N_1 + N_2 \cdot \mathbb{E}[\tau | \xi_0]$ , where  $N_1$  and  $N_2$  are the number of fitness evaluations for the initial population and each iteration, respectively. For example, for (1+1)-EA without noise,  $N_1 = 1$  and  $N_2 = 1$ . Note that, for EAs under noise, we assume that the reevaluation strategy [13,14,18] is used, i.e., when accessing the fitness of a solution, it is always reevaluated. For example, for (1+1)-EA with sampling, both  $f_k^N(x')$  and  $f_k^N(x)$  will be calculated and recalculated in each iteration; thus,  $N_1 = k$  and  $N_2 = 2k$ .

Lemma 1 characterizing the EFHT of a Markov chain by one-step transition and Lemma 2 showing the drift analysis tool will be used to analyze the EFHT of Markov chains in the paper. Drift analysis was first introduced to the running time analysis of EAs by He and Yao [16] and later many variants have been proposed (e.g., in [11,12]). To use it, a function  $V(x)$  has to be constructed to measure the distance of a state  $x$  to the optimal state space  $\mathcal{X}^*$ . The distance function  $V(x)$  satisfies that  $V(x \in \mathcal{X}^*) = 0$  and  $V(x \notin \mathcal{X}^*) > 0$ . Then, by investigating the progress on the distance to  $\mathcal{X}^*$  in each step, i.e.,  $\mathbb{E}[V(\xi_t) - V(\xi_{t+1}) | \xi_t]$ , an upper (lower) bound of the EFHT can be derived through dividing the initial distance by a lower (upper) bound of the progress.

**Lemma 1.** *Given a Markov chain  $\{\xi_t\}_{t=0}^{+\infty}$ , we have*

$$\begin{aligned} \forall x \in \mathcal{X}^* : \mathbb{E}[\tau | \xi_t = x] &= 0; \\ \forall x \notin \mathcal{X}^* : \mathbb{E}[\tau | \xi_t = x] &= 1 + \sum_{y \in \mathcal{X}} P(\xi_{t+1} = y | \xi_t = x) \mathbb{E}[\tau | \xi_{t+1} = y]. \end{aligned}$$

**Lemma 2 (Drift Analysis [16]).** *Given a Markov chain  $\{\xi_t\}_{t=0}^{+\infty}$  and a distance function  $V(x)$ , if it satisfies that for any  $t \geq 0$  and any  $\xi_t$  with  $V(\xi_t) > 0$ ,*

$$0 < c_l \leq \mathbb{E}[V(\xi_t) - V(\xi_{t+1}) | \xi_t] \leq c_u,$$

*then the EFHT satisfies that  $V(\xi_0)/c_u \leq \mathbb{E}[\tau | \xi_0] \leq V(\xi_0)/c_l$ .*

### 3 Theorem on Sampling Effectiveness

In this section, for EAs solving noisy problems, we provide two situations where the running time increases with the sample size, i.e., sampling is useless. Let  $\{\xi'_t\}_{t=0}^{+\infty}$  model the evolutionary process without noise, and let  $\{\xi_t\}_{t=0}^{+\infty}$  model that using the sample

size  $k$  for fitness evaluation under noise. We always denote  $\mathcal{X}$  and  $\mathcal{X}^*$  as the state space and the optimal state space, respectively. For any  $x, x' \in \mathcal{X}$ , let  $p(x, x')$  and  $q_k(x, x')$  denote the probability of jumping from state  $x$  to  $x'$  in one step for  $\{\xi'_t\}_{t=0}^{+\infty}$  and  $\{\xi_t\}_{t=0}^{+\infty}$  respectively, i.e.,  $p(x, x') = P(\xi'_{t+1} = x' | \xi'_t = x)$  and  $q_k(x, x') = P(\xi_{t+1} = x' | \xi_t = x)$ . For clarity, we also represent the EFHT of  $\{\xi'_t\}_{t=0}^{+\infty}$  and  $\{\xi_t\}_{t=0}^{+\infty}$  by  $\mathbb{E}[x]$  and  $\mathbb{E}_k[x]$  respectively, i.e.,  $\mathbb{E}[x] = \mathbb{E}[\tau' | \xi'_0 = x]$  and  $\mathbb{E}_k[x] = \mathbb{E}[\tau | \xi_0 = x]$ . Let  $\frac{dg(k)}{dk}$  denote the derivative of a function  $g(k)$  with respect to  $k$ .

**Theorem 1.** *For an EA  $\mathcal{A}$  optimizing a problem  $f$  under some kind of noise, if there exists a function  $g(k)$  ( $k \geq 1$ ) such that either one of the following two situations holds,*

$$(1) \max_{x \notin \mathcal{X}^*} \left\{ \sum_{x': \mathbb{E}[x'] \neq \mathbb{E}[x]} (q_k(x, x') - p(x, x')) (\mathbb{E}[x] - \mathbb{E}[x']) \right\} \leq g(k) < 0,$$

$$\text{and } 1 + g(k) - k \frac{dg(k)}{dk} \geq 0;$$

$$(2) \min_{x \notin \mathcal{X}^*} \left\{ \sum_{x': \mathbb{E}[x'] \neq \mathbb{E}[x]} (q_k(x, x') - p(x, x')) (\mathbb{E}[x] - \mathbb{E}[x']) \right\} \geq g(k) > 0, \text{ and } \frac{dg(k)}{dk} \leq 0,$$

then for any  $x \in \mathcal{X}$ ,  $k \cdot \mathbb{E}_k[x] \leq (k+1) \cdot \mathbb{E}_{k+1}[x]$ , i.e., sampling is useless.

Before the proof, we first intuitively explain these two situations where sampling is useless. In situation (1), noise is harmful and using a larger sample size may reduce its negative effect (i.e.,  $\mathbb{E}_k(x)$  decreases with  $k$ ), but the decrease rate of  $\mathbb{E}_k(x)$  is smaller than the increase rate of the sample size  $k$ ; thus sampling is overall useless. In situation (2), noise is actually helpful and using a larger sample size reduces its positive effect, thus  $\mathbb{E}_k(x)$  increases with  $k$  and sampling is of course useless.

**Proof.** We use Lemma 2 to prove a bound on  $\mathbb{E}_k[x]$ . We first construct a distance function  $\forall x \in \mathcal{X}$ ,  $V(x) = \mathbb{E}[x]$ , which satisfies that  $V(x \in \mathcal{X}^*) = 0$  and  $V(x \notin \mathcal{X}^*) > 0$  by Lemma 1. Then, we investigate  $\mathbb{E}[V(\xi_t) - V(\xi_{t+1}) | \xi_t = x]$  for any  $x \notin \mathcal{X}^*$ .

$$\begin{aligned} \mathbb{E}[V(\xi_t) - V(\xi_{t+1}) | \xi_t = x] &= V(x) - \mathbb{E}[V(\xi_{t+1}) | \xi_t = x] \\ &= 1 + \sum_{x' \in \mathcal{X}} p(x, x') \mathbb{E}[x'] - \sum_{x' \in \mathcal{X}} q_k(x, x') \mathbb{E}[x'] \quad (\text{by Lemma 1}) \\ &= 1 + \sum_{x': \mathbb{E}[x'] \neq \mathbb{E}[x]} (q_k(x, x') - p(x, x')) (\mathbb{E}[x] - \mathbb{E}[x']) =: 1 + g(x, k). \end{aligned}$$

If situation (1) holds,  $\mathbb{E}[V(\xi_t) - V(\xi_{t+1}) | \xi_t = x] \leq 1 + g(k)$ . By Lemma 2, we have  $\mathbb{E}_k[x] \geq \mathbb{E}[x] / (1 + g(k))$ , which shows that noise is harmful since  $g(k) < 0$ . The expected running time starting from  $x$  can be represented by  $Mk + Nk \cdot \mathbb{E}_k[x]$ , where  $M$  and  $N$  denote the number of solutions that need to be evaluated for the initial population and each iteration, respectively. The EFHT  $\mathbb{E}_k[x]$  may decrease with  $k$ ; thus we need to compare its decrease rate with the increase rate of the sample size  $k$ . The condition  $1 + g(k) - k \frac{dg(k)}{dk} \geq 0$  implies that  $\frac{d(k/(1+g(k)))}{dk} \geq 0$ , i.e., the decrease rate of  $\mathbb{E}_k[x]$  is smaller than the increase rate of  $k$ . Thus, the expected running time starting from  $x$  increases with  $k$ , i.e., sampling is useless.

If situation (2) holds,  $\mathbb{E}[V(\xi_t) - V(\xi_{t+1}) | \xi_t = x] \geq 1 + g(k)$ . By Lemma 2, we have  $\mathbb{E}_k[x] \leq \mathbb{E}[x] / (1 + g(k))$ , which shows that noise is helpful since  $g(k) > 0$ . Due to that  $\frac{dg(k)}{dk} \leq 0$  (i.e.,  $g(k)$  decreases with  $k$ ),  $\mathbb{E}_k[x]$  increases with  $k$ . Thus, the expected running time  $Mk + Nk \cdot \mathbb{E}_k[x]$  obviously increases with  $k$ , i.e., sampling is useless.  $\square$

## 4 Case Studies

In this section, we will apply the above theorem to analyze the effectiveness of sampling for EAs solving different pseudo-Boolean problems under additive Gaussian noise.

### 4.1 (1+1)-EA on Noisy OneMax

OneMax problem is to maximize the number of 1 bits of a solution  $x \in \{0, 1\}^n$ . It has become a benchmark for the running time analysis of EAs; particularly, the expected running time of (1+1)-EA with mutation probability  $\frac{1}{n}$  is  $\Theta(n \log n)$  [15]. For its noisy variant as in Definition 3, the fitness of a solution accessed in the optimization is a noisy one  $f^{\mathcal{N}}(x)$  instead of the true fitness  $f(x)$ .

**Definition 3 (Noisy OneMax Problem).** Given a distribution  $\mathcal{N}$  and  $n \in \mathbb{N}^+$ , defining

$$f^{\mathcal{N}}(x) = f(x) + \delta = \sum_{i=1}^n x_i + \delta$$

where  $x \in \{0, 1\}^n$  and  $\delta$  is randomly drawn from  $\mathcal{N}$ , Noisy OneMax Problem of size  $n$  is to solve the problem:  $\arg \max_{x \in \{0, 1\}^n} \mathbb{E}_{\delta \sim \mathcal{N}}[f^{\mathcal{N}}(x)]$ .

**Theorem 2.** For any  $\sigma > 0$ , sampling is useless for (1+1)-EA optimizing Noisy OneMax problem with Gaussian noise  $\mathcal{N} = N(\theta, \sigma^2)$ .

**Proof.** We are to show that the situation (1) of Theorem 1 holds here. From Lemma 1 in [21], we know that the EFHT  $\mathbb{E}[x]$  of (1+1)-EA on OneMax without noise depends on the number of 0 bits  $|x|_0$  and increases with it. Let  $mut(x, x')$  denote the probability of mutating from  $x$  to  $x'$  by step 3 of Algorithm 1. Note that, by sampling of size  $k$ , the Gaussian noise reduces to be  $N(\theta, \sigma^2/k)$ , i.e.,  $f_k^{\mathcal{N}}(x) = f(x) + \delta$  with  $\delta \sim N(\theta, \sigma^2/k)$ .

For any  $x$  with  $|x|_0 = i \geq 1$  and  $x'$  with  $|x'|_0 = j$ , if  $j < i$ ,  $p(x, x') = mut(x, x')$  since  $x'$  has less 0 bits and is better than  $x$ ; if  $j > i$ ,  $p(x, x') = 0$  since  $x'$  has more 0 bits and is worse than  $x$ . We also have  $q_k(x, x') = mut(x, x') \cdot Prob(f(x') + \delta_1 \geq f(x) + \delta_2)$ , where  $\delta_1, \delta_2 \sim N(\theta, \sigma^2/k)$ . Note that  $\delta_1 - \delta_2 \sim N(0, 2\sigma^2/k)$ . Thus,  $q_k(x, x') = mut(x, x') \cdot Prob(\delta \geq j - i)$ , where  $\delta \sim N(0, 2\sigma^2/k)$ . Then, we have

$$\begin{aligned} g(x, k) &= \sum_{x': \mathbb{E}[x'] \neq \mathbb{E}[x]} (q_k(x, x') - p(x, x')) (\mathbb{E}[x] - \mathbb{E}[x']) \\ &= - \sum_{|x'|_0 = j < i} mut(x, x') \cdot Prob(\delta > i - j) \cdot (\mathbb{E}[x] - \mathbb{E}[x']) \\ &\quad + \sum_{|x'|_0 = j > i} mut(x, x') \cdot Prob(\delta \geq j - i) \cdot (\mathbb{E}[x] - \mathbb{E}[x']) \\ &\leq -Prob(\delta > 1) \cdot \left( \sum_{|x'|_0 = i-1} mut(x, x') (\mathbb{E}[x] - \mathbb{E}[x']) \right). \end{aligned}$$

Let  $c = \min_{x \notin \mathcal{X}^*} \sum_{|x'|_0 = i-1} mut(x, x') (\mathbb{E}[x] - \mathbb{E}[x'])$ . Let  $\delta' \sim N(0, 1)$ . Then,  $Prob(\delta > 1) = Prob(\delta' > \frac{\sqrt{k}}{\sqrt{2\sigma}})$ . By  $Prob(\delta' > m) \geq \frac{m}{2\sqrt{2\pi}} e^{-m^2/2}$  for  $0 < m \leq 1$ ,

we can get  $Pr(\delta > 1) \geq \frac{\sqrt{k}}{4\sigma\sqrt{\pi}} \cdot e^{-k/4\sigma^2}$  when  $k \leq 2\sigma^2$ . Thus, let  $g(k) = -\frac{c\sqrt{k}}{4\sigma\sqrt{\pi}} \cdot e^{-k/4\sigma^2}$  which satisfies that  $\max_{x \notin \mathcal{X}^*} g(x, k) \leq g(k) < 0$ . Then,

$$1 + g(k) - k \frac{dg(k)}{dk} = 1 - \frac{c\sqrt{k}}{8\sigma\sqrt{\pi}} \cdot e^{-k/4\sigma^2} - \frac{ck\sqrt{k}}{16\sigma^3\sqrt{\pi}} \cdot e^{-k/4\sigma^2} \geq 1 - \frac{c}{2\sqrt{2\pi}}.$$

When  $k \geq 2\sigma^2$ ,  $Pr(\delta > 1) \approx \frac{\sigma}{\sqrt{k\pi}} \cdot e^{-k/4\sigma^2}$ , since  $Pr(\delta' > m) \approx \frac{1}{m} \cdot \frac{1}{\sqrt{2\pi}} e^{-m^2/2}$  for  $m \geq 1$ . Thus, let  $g(k) = -\frac{c\sigma}{\sqrt{k\pi}} \cdot e^{-k/4\sigma^2}$ . Then,

$$1 + g(k) - k \frac{dg(k)}{dk} = 1 - \frac{3c\sigma}{2\sqrt{k\pi}} \cdot e^{-k/4\sigma^2} - \frac{c\sqrt{k}}{4\sigma\sqrt{\pi}} \cdot e^{-k/4\sigma^2} \geq 1 - \frac{\sqrt{2}c}{\sqrt{\pi}} e^{-1/2},$$

where the inequality is since  $xe^{-x^2}$  reaches the maximum when  $x = \frac{\sqrt{2}}{2}$ . Then, we are to show that  $c \leq 1$ . By Lemma 1, for any  $x$  with  $|x|_0 = i \geq 1$ , we have

$$\begin{aligned} \mathbb{E}[x] &= 1 + \sum_{j=0}^{i-1} \sum_{|x'|_0=j} mut(x, x') \mathbb{E}[x'] + (1 - \sum_{j=0}^{i-1} \sum_{|x'|_0=j} mut(x, x')) \mathbb{E}[x] \\ &\leq 1 + \left( \sum_{j=0}^{i-1} \sum_{|x'|_0=j} mut(x, x') \right) \mathbb{E}[x' \mid |x'|_0 = i-1] + (1 - \sum_{j=0}^{i-1} \sum_{|x'|_0=j} mut(x, x')) \mathbb{E}[x]. \end{aligned}$$

Thus,  $\mathbb{E}[x] - \mathbb{E}[x' \mid |x'|_0 = i-1] \leq 1 / \sum_{j=0}^{i-1} \sum_{|x'|_0=j} mut(x, x')$ . Then,

$$\sum_{|x'|_0=i-1} mut(x, x') (\mathbb{E}[x] - \mathbb{E}[x']) \leq \sum_{|x'|_0=i-1} mut(x, x') / \sum_{j=0}^{i-1} \sum_{|x'|_0=j} mut(x, x') \leq 1,$$

which implies that  $c \leq 1$ . Thus,  $1 + g(k) - k \frac{dg(k)}{dk} \geq 0$ .  $\square$

## 4.2 (1+1)-EA on Noisy Trap

Trap problem is another commonly used problem in the theoretical analysis of EAs. It is to maximize the number of 0 bits of a solution except the global optimum 11...1; the expected running time of (1+1)-EA with mutation probability  $\frac{1}{n}$  is  $\Theta(n^n)$  [15].

**Definition 4 (Noisy Trap Problem).** Given a distribution  $\mathcal{N}$  and  $n \in \mathbb{N}^+$ , defining

$$f^{\mathcal{N}}(x) = f(x) + \delta = C \prod_{i=1}^n x_i - \sum_{i=1}^n x_i + \delta$$

where  $x \in \{0, 1\}^n$ ,  $C > n$  and  $\delta$  is randomly drawn from  $\mathcal{N}$ , Noisy Trap Problem of size  $n$  is to solve the problem:  $\arg \max_{x \in \{0, 1\}^n} \mathbb{E}_{\delta \sim \mathcal{N}} [f^{\mathcal{N}}(x)]$ .

**Theorem 3.** For any  $\sigma > 0$ , sampling is useless for (1+1)-EA optimizing Noisy Trap problem with Gaussian noise  $\mathcal{N} = N(\theta, \sigma^2)$  and  $C = +\infty$ .

**Proof.** We are to show that the situation (2) of Theorem 1 holds here. From Lemma 2 in [21], we know that the EFHT  $\mathbb{E}[x]$  of (1+1)-EA on Trap without noise depends on  $|x|_0$  and increases with it.

For any  $x$  with  $|x|_0 = i \geq 1$  and  $x'$  with  $|x'|_0 = j$ ,  $p(x, x') = 0$  if  $0 < j < i$ , and  $p(x, x') = mut(x, x')$  if  $j = 0$  or  $j > i$ ;  $q_k(x, x') = mut(x, x') \cdot Prob(\delta \geq i - j)$  if  $j > 0$  and  $q_k(x, x') = mut(x, x') \cdot Prob(\delta \geq i - C)$  if  $j = 0$ , where  $\delta \sim N(0, 2\sigma^2/k)$ . Note that  $C = +\infty$ , thus  $q_k(x, x') = mut(x, x')$  if  $j = 0$ . Then, we have

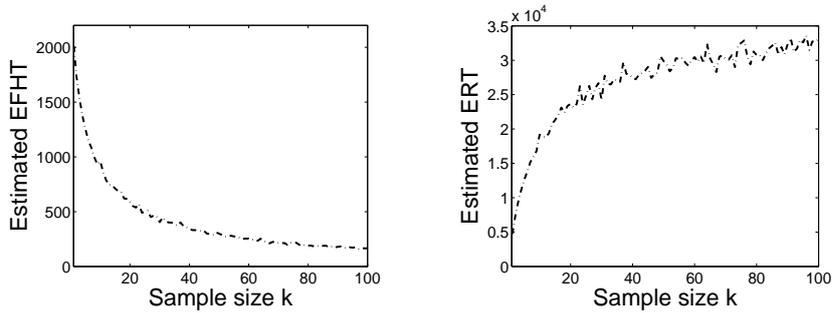
$$\begin{aligned} g(x, k) &= \sum_{x': \mathbb{E}[x'] \neq \mathbb{E}[x]} (q_k(x, x') - p(x, x')) (\mathbb{E}[x] - \mathbb{E}[x']) \\ &= \sum_{0 < |x'|_0 = j < i} mut(x, x') \cdot Prob(\delta \geq i - j) \cdot (\mathbb{E}[x] - \mathbb{E}[x']) \\ &\quad + \sum_{|x'|_0 = j > i} mut(x, x') \cdot Prob(\delta > j - i) \cdot (\mathbb{E}[x'] - \mathbb{E}[x]). \end{aligned}$$

Let  $g(k) = \min_{x \notin \mathcal{X}^*} g(x, k)$ , then  $g(k) > 0$ . When  $m > 0$ , we have that  $Prob(\delta \geq m)$  decreases with  $k$  by the property of Gaussian distribution, which implies that for any  $x$ ,  $g(x, k)$  decreases with  $k$ . Thus,  $g(k)$  decreases with  $k$ , i.e.,  $\frac{dg(k)}{dk} \leq 0$ .  $\square$

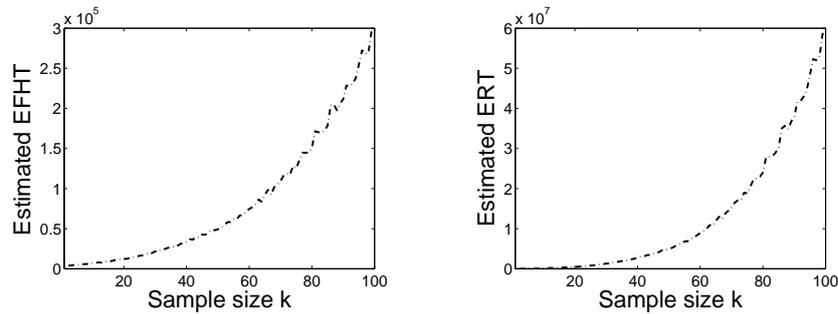
### 4.3 Empirical Verification

We run (1+1)-EA on the problems to verify the theoretical results. For the (1+1)-EA, the mutation probability  $p$  is set to be  $\frac{1}{n}$ ; for the OneMax and the Trap problems, the problem size  $n = 10$  and  $C = n + 1$ ; for the Gaussian noise,  $\theta = 0$  and  $\sigma = 10$ . We investigate the sample size  $k$  from 1 to 100; for each  $k$ , we run the EA 1,000 times independently, where each run stops until an optimal solution is found. We use the average number of iterations and the average number of fitness evaluations as the estimation of the EFHT and the expected running time (ERT), respectively.

The results are plotted in Figures 1 and 2. For (1+1)-EA optimizing Noisy OneMax problem, Figure 1 shows that the EFHT can decrease by increasing the sample size  $k$ , however the ERT increases with  $k$ , which implies that the decrease rate of the EFHT cannot catch up with the increase rate of  $k$ . On Noisy Trap problem, we can observe from Figure 2 that both the EFHT and ERT increase with the sample size, which implies that noise is helpful and using a larger sample size reduces its positive effect. Thus, these empirical results verify our theoretical analysis.



**Figure 1.** Estimated EFHT and ERT for (1+1)-EA on Noisy OneMax with different sample sizes.



**Figure 2.** Estimated EFHT and ERT for (1+1)-EA on Noisy Trap with different sample sizes.

## 5 Conclusion

Sampling has often been employed to smooth noise in evolutionary optimization. Previous empirical studies showed conflicting results, and sampling thus has not been well understood. In this paper, we investigate its effectiveness by rigorous running time analysis. We provide a sufficient condition under which sampling is useless. Using this condition, we prove that sampling is useless for (1+1)-EA optimizing OneMax and Trap problems under additive Gaussian noise, which is also empirically verified. An intuitive interpretation of the theorems is that, noise should be removed for the OneMax problem, but the extra cost of using sampling is overwhelming; and noise should not be removed for the Trap problem, thus sampling is useless. Note that, OneMax and Trap have been recognized as the easiest and the hardest instances, respectively, in the pseudo-Boolean problem class with a unique global optimum for (1+1)-EA [21]. Thus, we conjecture that sampling might be useless for a large problem class, which will be a subject of future research. Our results on the effectiveness of sampling may guide us to design effective noise handling strategies in real optimization tasks.

## References

1. Aizawa, A.N., Wah, B.W.: Scheduling of genetic algorithms in a noisy environment. *Evolutionary Computation* 2(2), 97–122 (1994)
2. Arnold, D.V.: *Noisy Optimization with Evolution Strategies*. Kluwer, Norwell, MA (2002)
3. Auger, A., Doerr, B.: *Theory of Randomized Search Heuristics: Foundations and Recent Developments*. World Scientific, Singapore (2011)
4. Bäck, T.: *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford University Press, Oxford, UK (1996)
5. Bartz-Beielstein, T., Blum, D., Branke, J.: Particle swarm optimization and sequential sampling in noisy environments. In: *Metaheuristics, Operations Research/Computer Science Interfaces Series*, vol. 39, pp. 261–273. Springer, Heidelberg (2007)
6. Beyer, H.G.: Evolutionary algorithms in noisy environments: theoretical issues and guidelines for practice. *Computer Methods in Applied Mechanics and Engineering* 186(2), 239–267 (2000)
7. Branke, J., Schmidt, C.: Selection in the presence of noise. In: *Proceedings of the 5th ACM Conference on Genetic and Evolutionary Computation*. pp. 766–777. Chicago, IL (2003)

8. Branke, J., Schmidt, C.: Sequential sampling in noisy environments. In: Proceedings of the 8th International Conference on Parallel Problem Solving from Nature. pp. 202–211. Birmingham, UK (2004)
9. Branke, J., Schmidt, C., Schmeck, H.: Efficient fitness estimation in noisy environments. In: Proceedings of the 3rd ACM Conference on Genetic and Evolutionary Computation. pp. 243–250. San Francisco, CA (2001)
10. Cantú-Paz, E.: Adaptive sampling for noisy problems. In: Proceedings of the 6th ACM Conference on Genetic and Evolutionary Computation. pp. 947–958. Seattle, WA (2004)
11. Doerr, B., Goldberg, L.A.: Adaptive drift analysis. *Algorithmica* 65, 224–250 (2013)
12. Doerr, B., Johannsen, D., Winzen, C.: Multiplicative drift analysis. *Algorithmica* 64, 673–697 (2012)
13. Doerr, B., Hota, A., Kötzing, T.: Ants easily solve stochastic shortest path problems. In: Proceedings of the 14th ACM Conference on Genetic and Evolutionary Computation. pp. 17–24. Philadelphia, PA (2012)
14. Droste, S.: Analysis of the (1+1) EA for a noisy OneMax. In: Proceedings of the 6th ACM Conference on Genetic and Evolutionary Computation. pp. 1088–1099. Seattle, WA (2004)
15. Droste, S., Jansen, T., Wegener, I.: On the analysis of the (1+1) evolutionary algorithm. *Theoretical Computer Science* 276(1-2), 51–81 (2002)
16. He, J., Yao, X.: Drift analysis and average time complexity of evolutionary algorithms. *Artificial Intelligence* 127(1), 57–85 (2001)
17. Iacca, G., Neri, F., Mininno, E.: Noise analysis compact differential evolution. *International Journal of Systems Science* 43(7), 1248–1267 (2012)
18. Jin, Y., Branke, J.: Evolutionary optimization in uncertain environments—a survey. *IEEE Transactions on Evolutionary Computation* 9(3), 303–317 (2005)
19. Neumann, F., Witt, C.: *Bioinspired Computation in Combinatorial Optimization: Algorithms and Their Computational Complexity*. Springer-Verlag, Berlin, Germany (2010)
20. Park, T., Ryu, K.R.: Accumulative sampling for noisy evolutionary multi-objective optimization. In: Proceedings of the 13th ACM Conference on Genetic and Evolutionary Computation. pp. 793–800. Dublin, Ireland (2011)
21. Qian, C., Yu, Y., Zhou, Z.H.: On algorithm-dependent boundary case identification for problem classes. In: Proceedings of the 12th International Conference on Parallel Problem Solving from Nature. pp. 62–71. Taormina, Italy (2012)
22. Sano, Y., Kita, H.: Optimization of noisy fitness functions by means of genetic algorithms using history of search with test of estimation. In: Proceedings of the 2002 IEEE Congress on Evolutionary Computation. pp. 360–365. Honolulu, HI (2002)
23. Siegmund, F., Ng, A.H., Deb, K.: A comparative study of dynamic resampling strategies for guided evolutionary multi-objective optimization. In: Proceedings of the 2013 IEEE Congress on Evolutionary Computation. pp. 1826–1835. Cancun, Mexico (2013)
24. Stagge, P.: Averaging efficiently in the presence of noise. In: Proceedings of the 5th International Conference on Parallel Problem Solving from Nature. pp. 188–197. Amsterdam, The Netherlands (1998)
25. Syberfeldt, A., Ng, A., John, R.I., Moore, P.: Evolutionary optimisation of noisy multi-objective problems using confidence-based dynamic resampling. *European Journal of Operational Research* 204(3), 533–544 (2010)
26. Yu, Y., Zhou, Z.H.: A new approach to estimating the expected first hitting time of evolutionary algorithms. *Artificial Intelligence* 172(15), 1809–1832 (2008)
27. Zhang, Z., Xin, T.: Immune algorithm with adaptive sampling in noisy environments and its application to stochastic optimization problems. *IEEE Computational Intelligence Magazine* 2(4), 29–40 (2007)