A Lower Bound Analysis of Population-based Evolutionary Algorithms for Pseudo-Boolean Functions[*]

Chao Qian[1,2], Yang Yu[2], and Zhi-Hua Zhou[2]

[1]UBRI, School of Computer Science and Technology,
University of Science and Technology of China, Hefei 230027, China
[2]National Key Laboratory for Novel Software Technology,
Nanjing University, Nanjing 210023, China
`chaoqian@ustc.edu.cn`, `{yuy,zhouzh}@nju.edu.cn`

Abstract Evolutionary algorithms (EAs) are population-based general-purpose optimization algorithms, and have been successfully applied in real-world optimization tasks. However, previous theoretical studies often employ EAs with only a parent or offspring population and focus on specific problems. Furthermore, they often only show upper bounds on the running time, while lower bounds are also necessary to get a complete understanding of an algorithm. In this paper, we analyze the running time of the $(\mu+\lambda)$-EA (a general population-based EA with mutation only) on the class of pseudo-Boolean functions with a unique global optimum. By applying the recently proposed switch analysis approach, we prove the lower bound $\Omega(n \ln n + \mu + \lambda n \ln \ln n / \ln n)$ for the first time. Particularly on the two widely-studied problems, OneMax and LeadingOnes, the derived lower bound reveals that the $(\mu+\lambda)$-EA will be slower than the (1+1)-EA when the population size μ or λ is above a moderate order. Our results imply that the increase of population size, while usually desired in practice, bears the risk of increasing the lower bound of the running time and thus should be carefully considered.

1 Introduction

Evolutionary algorithms (EAs) [2] are a kind of population-based heuristic optimization algorithm. They have been widely applied in industrial optimization problems. However, the theoretical analysis is difficult due to their complexity and randomness. In the recent decade, there has been a significant rise on the running time analysis (one essential theoretical aspect) of EAs [1,16]. For example, Droste et al. [7] proved that the expected running time of the (1+1)-EA on linear pseudo-Boolean functions is $\Theta(n \ln n)$; for the $(\mu+1)$-EA solving several artificially designed functions, a large parent population size μ was shown to be able to reduce the running time from exponential to polynomial [13,17,19,20]; for the $(1+\lambda)$-EA solving linear functions, the expected running time was proved to be $O(n \ln n + \lambda n)$ [6], and a tighter bound up to lower order terms was derived on the specific linear function OneMax [10].

Previous running time analyses often consider EAs with only a parent or offspring population, which do not fully reflect the population-based nature of real EAs. When involving both parent and offspring populations, the running time analysis gets more complex, and only a few results have been reported on the $(\lambda+\lambda)$-EA (i.e., a specific version of the $(\mu+\lambda)$-EA with $\mu = \lambda$), which maintains λ solutions and generates λ offspring solutions by only mutation in each iteration. He and Yao [12] compared the expected running time of the (1+1)-EA and the $(\lambda+\lambda)$-EA on two specific artificial problems, and proved that the introduction of a population can reduce the running time exponentially. On the contrary side, Chen et al. [5] found that a large population size is harmful for the $(\lambda+\lambda)$-EA solving the TrapZeros problem. Chen et al. [4] also proved that the expected running time of the $(\lambda+\lambda)$-EA on the OneMax and LeadingOnes problems is $O(\lambda n \ln \ln n + n \ln n)$ and $O(\lambda n \ln \ln n + n^2)$, respectively. Later, a low selection pressure was shown to be better for the $(\lambda+\lambda)$-EA solving a wide gap problem [3], and a proper mutation-selection balance was proved to be necessary for the effectiveness of the $(\lambda+\lambda)$-EA solving the SelPres problem [15].

The above-mentioned studies on the $(\lambda+\lambda)$-EA usually focus on specific test functions, while EAs are general purpose optimization algorithms and can be applied to all optimization problems where solutions can be represented and evaluated. Thus, it is necessary to analyze EAs over large problem classes. Meanwhile, most previous running time analyses on population-based EAs only show upper bounds. Although upper bounds are appealing for revealing the ability of an algorithm, lower bounds which reveal the limitation are also necessary for a complete understanding of the algorithm.

In this paper, we analyze the running time of the $(\mu+\lambda)$-EA solving the class of *pseudo-Boolean functions with a unique global optimum*, named `UBoolean`, which covers many P and NP-hard combinatorial problems. By applying the recently proposed approach switch analysis [21], we prove that the expected running time is lower bounded by $\Omega(n \ln n + \mu + \lambda n \ln \ln n / \ln n)$. Particularly, when applying this lower bound to the two specific problems, OneMax and LeadingOnes, we can have a more complete understanding of the impact of the offspring population size λ. It was known that the $(\mu+\lambda)$-EA is always not asymptotically faster than the (1+1)-EA on these two problems [14,18]. But it was left open that what is the range of λ where the $(\mu+\lambda)$-EA is asymptotically slower than the (1+1)-EA. Note that the expected running time of the (1+1)-EA on OneMax and LeadingOnes is $\Theta(n \ln n)$ and $\Theta(n^2)$, respectively [7]. Thus, we can easily get that the $(\mu+\lambda)$-EA is asymptotically slower than the (1+1)-EA when $\lambda \in \omega(\frac{(\ln n)^2}{\ln \ln n})$ on OneMax and $\lambda \in \omega(\frac{n \ln n}{\ln \ln n})$ on LeadingOnes. For the parent population size μ, we easily get obvious ranges $\omega(n \ln n)$ and $\omega(n^2)$ for the $(\mu+\lambda)$-EA being asymptotically slower on OneMax and LeadingOnes, respectively.

The rest of this paper is organized as follows. Section 2 introduces some preliminaries. Section 3 introduces the employed analysis approach. The running time analysis of the $(\mu+\lambda)$-EA on `UBoolean` is presented in Section 4. Section 5 concludes the paper.

2 Preliminaries

In this section, we first introduce the $(\mu+\lambda)$-EA and the pseudo-Boolean problem class studied in this paper, respectively, then describe how to model EAs as Markov chains.

Algorithm 1 $(\mu+\lambda)$-EA

Given solution length n and objective function f, let every population, denoted by variable ξ, contain μ solutions. The $(\mu+\lambda)$-EA consists of the following steps:

1: let $t \leftarrow 0$, and $\xi_0 \leftarrow \mu$ solutions uniformly and randomly selected from $\{0,1\}^n$.
2: **repeat until** some criterion is met
3:     **for** $i = 1$ to λ
4:         select a solution s from ξ_t according to some selection mechanism.
5:         create s_i' by flipping each bit of s with probability $1/n$.
6:     **end for**
7:     $\xi_{t+1} :=$ select μ solutions from $\xi_t \cup \{s_1', \ldots, s_\lambda'\}$ according to some strategy.
8:     let $t \leftarrow t + 1$.

2.1 $(\mu+\lambda)$-EA

EAs [2] are used as general heuristic randomized optimization approaches. Starting from an initial set of solutions (called a population), EAs try to 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 $(\mu+\lambda)$-EA as described in Algorithm 1 is a general population-based EA with mutation only for optimizing pseudo-Boolean problems over $\{0,1\}^n$. It maintains μ solutions. In each iteration, one solution selected from the current population is used to generate an offspring solution by bit-wise mutation (i.e., line 5); this process is repeated independently for λ times; then μ solutions out of the parent and offspring solutions are selected to be the next population. Note that the selection strategies for reproducing new solutions and updating the population can be arbitrary. Thus, the considered $(\mu+\lambda)$-EA is quite general, and covers most population-based EAs with mutation only in previous theoretical analyses, e.g., [5,12,15].

The running time of EAs is usually defined as the number of fitness evaluations until an optimal solution is found for the first time, since the fitness evaluation is the computational process with the highest cost of the algorithm [11,22]. Note that running time analysis has been a leading theoretical aspect for randomized search heuristics [1,16].

2.2 Pseudo-Boolean Function Problems

The pseudo-Boolean function class is a large function class which only requires the solution space to be $\{0,1\}^n$ and the objective space to be $\mathbb{R}$. It covers many typical P and NP-hard combinatorial problems such as minimum spanning tree and minimum set cover. We consider a subclass named `UBoolean` as shown in Definition 1, in which every function has a unique global optimum. Note that maximization is considered since minimizing f is equivalent to maximizing $-f$. For any function in `UBoolean`, we assume without loss of generality that the optimal solution is $11 \ldots 1$ (briefly denoted as 1^n). This is because EAs treat the bits 0 and 1 symmetrically, and thus the 0-bits in an optimal solution can be interpreted as 1-bits without affecting the behavior of EAs. The expected running time of unbiased black-box algorithms and mutation-based EAs on `UBoolean` has been proved to be $\Omega(n \ln n)$ [14,18].

Definition 1 (`UBoolean`). *A function $f : \{0,1\}^n \to \mathbb{R}$ in `UBoolean` satisfies that*

$$\exists s \in \{0,1\}^n, \forall s' \in \{0,1\}^n - \{s\}, f(s') < f(s).$$

Diverse pseudo-Boolean problems in `UBoolean` have been used for analyzing the running time of EAs, and then to disclose properties of EAs. Here, we introduce the LeadingOnes problem, which will be used in this paper. As presented in Definition 2, it is to maximize the number of consecutive 1-bits starting from the left. It has been proved that the expected running time of the (1+1)-EA on LeadingOnes is $\Theta(n^2)$ [7].

Definition 2 (LeadingOnes). *The LeadingOnes Problem of size n is to find an n bits binary string s^* such that, letting s_j be the j-th bit of a solution $s \in \{0,1\}^n$,*

$$s^* = \arg\max_{s\in\{0,1\}^n} \left(f(s) = \sum_{i=1}^{n} \prod_{j=1}^{i} s_j \right).$$

2.3 Markov Chain Modeling

EAs can be modeled and analyzed as Markov chains, e.g., in [11,22]. Let $\mathcal{X}$ be the population space and $\mathcal{X}^* \subseteq \mathcal{X}$ be the optimal population space. Note that an optimal population in $\mathcal{X}^*$ contains at least one optimal solution. Let $\xi_t \in \mathcal{X}$ be the population after t generations. Then, an EA can be described as a random sequence $\{\xi_0, \xi_1, \xi_2, \ldots\}$. Since ξ_{i+1} can often be decided from ξ_i and the reproduction operator of the EA (i.e., $P(\xi_{i+1} \mid \xi_i, \xi_{i-1}, \ldots, \xi_0) = P(\xi_{i+1} \mid \xi_i)$), the random sequence forms a Markov chain $\{\xi_t\}_{t=0}^{+\infty}$ with state space $\mathcal{X}$, denoted as "$\xi \in \mathcal{X}$" for simplicity. Note that all sets considered in this paper are multisets, e.g., a population can contain several copies of the same solution.

The goal of EAs is to reach the optimal space $\mathcal{X}^*$ from an initial population ξ_0. Given a Markov chain $\xi \in \mathcal{X}$ modeling an EA and $t_0 \geq 0$, we define τ as a random variable such that $\tau = \min\{t \geq 0 \mid \xi_{t_0+t} \in \mathcal{X}^*\}$. That is, τ is the number of steps needed to reach the optimal space for the first time when starting from time t_0. The mathematical expectation of τ, $\mathbb{E}[\![\tau \mid \xi_{t_0} = x]\!] = \sum_{i=0}^{\infty} iP(\tau = i)$, is called the *conditional first hitting time* (CFHT) of the chain staring from $\xi_{t_0} = x$. If ξ_{t_0} is drawn from a distribution π_{t_0}, the expectation of the CFHT over π_{t_0}, $\mathbb{E}[\![\tau \mid \xi_{t_0} \sim \pi_{t_0}]\!] = \sum_{x\in\mathcal{X}} \pi_{t_0}(x)\mathbb{E}[\![\tau \mid \xi_{t_0} = x]\!]$, is called the *distribution-CFHT* (DCFHT) of the chain from $\xi_{t_0} \sim \pi_{t_0}$. Since the running time of EAs is counted by the number of fitness evaluations, the cost of initialization and each generation should be considered. For example, the expected running time of the ($\mu+\lambda$)-EA is $\mu + \lambda \cdot \mathbb{E}[\![\tau \mid \xi_0 \sim \pi_0]\!]$.

A Markov chain $\xi \in \mathcal{X}$ is said to be absorbing, if $\forall t \geq 0 : P(\xi_{t+1} \in \mathcal{X}^* \mid \xi_t \in \mathcal{X}^*) = 1$. Note that all Markov chains modeling EAs can be transformed to be absorbing by making it unchanged once an optimal state has been found. This transformation obviously does not affect its first hitting time.

3 The Switch Analysis Approach

To derive running time bounds of the ($\mu+\lambda$)-EA on `UBoolean`, we first model the EA process as a Markov chain, and then apply the switch analysis approach.

Switch analysis [21,23] as presented in Theorem 1 is a recently proposed approach that compares the DCFHT of two Markov chains. Since the state spaces of the two chains may be different, an aligned mapping function $\phi : \mathcal{X} \to \mathcal{Y}$ as shown in Definition 3 is employed. Note that $\phi^{-1}(y) = \{x \in \mathcal{X} \mid \phi(x) = y\}$. Using switch analysis to derive running time bounds of a given chain $\xi \in \mathcal{X}$ (i.e., modeling a given EA running on a given problem), one needs to

1. construct a reference chain $\xi' \in \mathcal{Y}$ for comparison and design an aligned mapping function ϕ from $\mathcal{X}$ to $\mathcal{Y}$;
2. analyze their one-step transition probabilities, i.e., $P(\xi_{t+1} \mid \xi_t)$ and $P(\xi'_{t+1} \mid \xi'_t)$, the CFHT of the chain $\xi' \in \mathcal{Y}$, i.e., $\mathbb{E}[\![\tau' \mid \xi'_t]\!]$, and the state distribution of the chain $\xi \in \mathcal{X}$, i.e., π_t;
3. examine Eq. (1) to get the difference ρ_t between each step of the two chains;
4. sum up ρ_t to get a running time gap ρ of the two chains, and then bounds on $\mathbb{E}[\![\tau \mid \xi_0]\!]$ can be derived by combining $\mathbb{E}[\![\tau' \mid \xi'_0]\!]$ with ρ.

Definition 3 (Aligned Mapping [21]). *Given two spaces $\mathcal{X}$ and $\mathcal{Y}$ with target subspaces $\mathcal{X}^*$ and $\mathcal{Y}^*$, respectively, a function $\phi : \mathcal{X} \to \mathcal{Y}$ is called*
(a) a left-aligned mapping *if* $\forall x \in \mathcal{X}^* : \phi(x) \in \mathcal{Y}^*$;
(b) a right-aligned mapping *if* $\forall x \in \mathcal{X} - \mathcal{X}^* : \phi(x) \notin \mathcal{Y}^*$;
(c) an optimal-aligned mapping *if it is both left-aligned and right-aligned.*

Theorem 1 (Switch Analysis [21]). *Given two absorbing Markov chains $\xi \in \mathcal{X}$ and $\xi' \in \mathcal{Y}$, let τ and τ' denote the hitting events of ξ and ξ', respectively, and let π_t denote the distribution of ξ_t. Given a series of values $\{\rho_t \in \mathbb{R}\}_{t=0}^{+\infty}$ with $\rho = \sum_{t=0}^{+\infty} \rho_t$ and a right (or left)-aligned mapping $\phi : \mathcal{X} \to \mathcal{Y}$, if $\mathbb{E}[\![\tau \mid \xi_0 \sim \pi_0]\!]$ is finite and*

$$
\begin{aligned}
\forall t : \quad &\sum_{x \in \mathcal{X}, y \in \mathcal{Y}} \pi_t(x) P(\xi_{t+1} \in \phi^{-1}(y) \mid \xi_t = x) \mathbb{E}[\![\tau' \mid \xi'_0 = y]\!] \\
&\leq (or \ \geq) \sum_{u, y \in \mathcal{Y}} \pi_t^{\phi}(u) P(\xi'_1 = y \mid \xi'_0 = u) \mathbb{E}[\![\tau' \mid \xi'_1 = y]\!] + \rho_t,
\end{aligned}
\tag{1}
$$

where $\pi_t^{\phi}(u) = \pi_t(\phi^{-1}(u)) = \sum_{x \in \phi^{-1}(u)} \pi_t(x)$, we have

$$
\mathbb{E}[\![\tau \mid \xi_0 \sim \pi_0]\!] \leq (or \ \geq) \mathbb{E}[\![\tau' \mid \xi'_0 \sim \pi_0^{\phi}]\!] + \rho.
$$

The idea of switch analysis is to obtain the difference ρ on the DCFHT of two chains by summing up all the one-step differences ρ_t. Using Theorem 1 to compare two chains, we can waive the long-term behavior of one chain, since Eq. (1) does not involve the term $\mathbb{E}[\![\tau \mid \xi_t]\!]$. Therefore, the theorem can simplify the analysis of an EA process by comparing it with an easy-to-analyze one.

4 Running Time Analysis

In this section, we prove a lower bound on the expected running time of the $(\mu+\lambda)$-EA solving `UBoolean`, as shown in Theorem 2. Our proof is accomplished by using switch

analysis (i.e., Theorem 1). The target EA process we are to analyze is the $(\mu+\lambda)$-EA running on any function in `UBoolean`. The constructed reference process for comparison is the $\text{RLS}^{\neq}$ algorithm running on the LeadingOnes problem. $\text{RLS}^{\neq}$ is a modification of the randomized local search algorithm. It maintains only one solution s. In each iteration, a new solution s' is generated by flipping a randomly chosen bit of s, and s' is accepted only if $f(s') > f(s)$. That is, $\text{RLS}^{\neq}$ searches locally and only accepts a better offspring solution.

Theorem 2. *The expected running time of the $(\mu+\lambda)$-EA on* `UBoolean` *is* $\Omega(n \ln n + \mu + \lambda n \ln \ln n / \ln n)$, *when μ and λ are upper bounded by a polynomial in n.*

We first give some lemmas that will be used in the proof of Theorem 2. Lemma 1 characterizes the one-step transition behavior of a Markov chain via CFHT. Lemma 2 gives the CFHT $\mathbb{E}[\![\tau' \mid \xi'_t = y]\!]$ of the reference chain ξ' (i.e., $\text{RLS}^{\neq}$ running on LeadingOnes). In the following analysis, we will use $\mathbb{E}_{rls}(j)$ to denote $\mathbb{E}[\![\tau' \mid \xi'_t = y]\!]$ with $|y|_0 = j$, i.e., $\mathbb{E}_{rls}(j) = nj$.

Lemma 1 ([9]). *Given a Markov chain $\xi \in \mathcal{X}$ and a target subspace $\mathcal{X}^* \subset \mathcal{X}$, we have, for CFHT,* $\forall x \in \mathcal{X}^* : \mathbb{E}[\![\tau \mid \xi_t = x]\!] = 0$,

$$\forall x \notin \mathcal{X}^* : \mathbb{E}[\![\tau \mid \xi_t = x]\!] = 1 + \sum_{x' \in \mathcal{X}} P(\xi_{t+1} = x' \mid \xi_t = x) \mathbb{E}[\![\tau \mid \xi_{t+1} = x']\!].$$

Lemma 2 ([21]). *For the chain $\xi' \in \mathcal{Y}$ modeling $\text{RLS}^{\neq}$ running on the LeadingOnes problem, the CFHT satisfies that $\forall y \in \mathcal{Y} = \{0,1\}^n : \mathbb{E}[\![\tau' \mid \xi'_t = y]\!] = n \cdot |y|_0$, where $|y|_0$ denotes the number of 0-bits of y.*

Lemma 3. *For $m \geq i \geq 0$, $\sum_{k=0}^{i} \binom{m}{k}(\frac{1}{n})^k(1 - \frac{1}{n})^{m-k}$ decreases with m.*

Proof. Let $f(m) = \sum_{k=0}^{i} \binom{m}{k}(\frac{1}{n})^k(1 - \frac{1}{n})^{m-k}$. The goal is to show that $f(m+1) \leq f(m)$ for $m \geq i$. Denote $X_1, \ldots, X_{m+1}$ as independent random variables, where X_j satisfies that $P(X_j = 1) = \frac{1}{n}$ and $P(X_j = 0) = 1 - \frac{1}{n}$. Then we can express $f(m)$ and $f(m+1)$ as $f(m) = P(\sum_{j=1}^{m} X_j \leq i)$ and $f(m+1) = P(\sum_{j=1}^{m+1} X_j \leq i)$. Thus,

$$f(m+1) = P(\sum_{j=1}^{m} X_j < i) + P(\sum_{j=1}^{m} X_j = i)P(X_{m+1} = 0)$$

$$= P(\sum_{j=1}^{m} X_j < i) + P(\sum_{j=1}^{m} X_j = i)(1 - \frac{1}{n}) \leq f(m). \qquad \square$$

Lemma 4. *For $\lambda \leq n^c$ where c is a positive constant, it holds that*

$$\sum_{i=0}^{n-1} \left(\sum_{k=0}^{i} \binom{n}{k}(\frac{1}{n})^k(1 - \frac{1}{n})^{n-k} \right)^{\lambda} \geq n - \left\lceil \frac{e(c+1)\ln n}{\ln \ln n} \right\rceil.$$

Proof. Let $m = \lceil \frac{e(c+1)\ln n}{\ln \ln n} \rceil$. Denote $X_1, ..., X_n$ as independent random variables, where $P(X_j = 1) = \frac{1}{n}$ and $P(X_j = 0) = 1 - \frac{1}{n}$. Let $X = \sum_{j=1}^{n} X_j$, then its expectation $\mathbb{E}[\![X]\!] = 1$. We thus have

$$\forall i \geq m, \sum_{k=i}^{n} \binom{n}{k}(\frac{1}{n})^k(1 - \frac{1}{n})^{n-k} = P(X \geq i) \leq e^{(i-1)}/i^i, \qquad (2)$$

where the inequality is by Chernoff bound. Then, we have

$$\sum_{i=0}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda \geq \sum_{i=m-1}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda$$

$$= \sum_{i=m-1}^{n-1}(1-\sum_{k=i+1}^{n}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda$$

$$\geq \sum_{i=m-1}^{n-1}(1-e^i/(i+1)^{(i+1)})^\lambda \geq \sum_{i=m-1}^{n-1}(1-e^{m-1}/m^m)^\lambda,$$

where the second inequality is by Eq. (2), and the last inequality can be easily derived because $e^i/(i+1)^{(i+1)}$ decreases with i when $i \geq m-1$.

Then, we evaluate e^{m-1}/m^m by taking logarithm to its reciprocal.

$$\ln(m^m/e^{m-1}) = m(\ln m - 1) + 1$$

$$\geq \frac{e(c+1)\ln n}{\ln\ln n}(1+\ln(c+1)+\ln\ln n - \ln\ln\ln n - 1) + 1$$

$$\geq \frac{e(c+1)\ln n}{\ln\ln n}\frac{1}{e}\ln\ln n = (c+1)\ln n \geq \ln\lambda n. \quad \text{(by } \lambda \leq n^c)$$

This implies that $e^{m-1}/m^m \leq \frac{1}{\lambda n}$. Thus, we have

$$\sum_{i=0}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda \geq \sum_{i=m-1}^{n-1}(1-\frac{1}{\lambda n})^\lambda$$

$$\geq \sum_{i=m-1}^{n-1}(1-\frac{1}{n}) \geq n-m = n - \lceil\frac{e(c+1)\ln n}{\ln\ln n}\rceil,$$

where the second inequality is by $\forall 0 \leq xy \leq 1, y \geq 1 : (1-x)^y \geq 1-xy.$ □

Lemma 5 ([8]). *Let* $H(\epsilon) = -\epsilon\log\epsilon - (1-\epsilon)\log(1-\epsilon)$. *It holds that*

$$\forall n \geq 1, 0 < \epsilon < \frac{1}{2} : \sum_{k=0}^{\lfloor\epsilon n\rfloor}\binom{n}{k} \leq 2^{H(\epsilon)n}.$$

Proof of Theorem 2. We use switch analysis (i.e., Theorem 1) to prove it. Let $\xi \in \mathcal{X}$ model the analyzed EA process (i.e., the $(\mu+\lambda)$-EA running on any function in UBoolean). We use RLS$^{\neq}$ running on the LeadingOnes problem as the reference process modeled by $\xi' \in \mathcal{Y}$. Then, $\mathcal{Y} = \{0,1\}^n$, $\mathcal{X} = \{\{y_1, y_2, \ldots, y_\mu\} \mid y_i \in \{0,1\}^n\}$, $\mathcal{Y}^* = \{1^n\}$ and $\mathcal{X}^* = \{x \in \mathcal{X} \mid \max_{y\in x}|y|_1 = n\}$, where $|y|_1$ denotes the number of 1-bits of a solution $y \in \{0,1\}^n$. We construct a mapping $\phi : \mathcal{X} \to \mathcal{Y}$ as that $\forall x \in \mathcal{X} : \phi(x) = \arg\max_{y\in x}|y|_1$. It is easy to see that the mapping is an optimality-aligned mapping, because $\phi(x) \in \mathcal{Y}^*$ iff $x \in \mathcal{X}^*$.

We investigate the condition Eq. (1) of switch analysis. For any $x \notin \mathcal{X}^*$, suppose that $\min\{|y|_0 \mid y \in x\} = j > 0$. Then, $|\phi(x)|_0 = j$. By Lemma 1 and 2, we have

$$\sum_{y\in\mathcal{Y}}P(\xi_1' = y \mid \xi_0' = \phi(x))\mathbb{E}[\![\tau' \mid \xi_1' = y]\!] = \mathbb{E}_{rls}(j) - 1 = nj - 1. \qquad (3)$$

For the reproduction of the ($\mu+\lambda$)-EA (i.e., the chain $\xi \in \mathcal{X}$) on the population x, assume that the λ selected solutions from x for reproduction have the number of 0-bits $j_1, j_2, ..., j_\lambda$, respectively, where $j \leq j_1 \leq j_2 \leq ... \leq j_\lambda \leq n$. If there are at most i ($0 \leq i \leq j_1$) number of 0-bits mutating to 1-bits for each selected solution and there exists at least one selected solution which flips exactly i number of 0-bits, which happens with probability $\prod_{p=1}^{\lambda}(\sum_{k=0}^{i}\binom{j_p}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{j_p-k}) - \prod_{p=1}^{\lambda}(\sum_{k=0}^{i-1}\binom{j_p}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{j_p-k})$ (denoted by $p(i)$), the next population x' satisfies that $|\phi(x')|_0 \geq j_1 - i$. Furthermore, $\mathbb{E}_{rls}(i) = ni$ increases with i. Thus, we have

$$\sum_{y \in \mathcal{Y}} P(\xi_{t+1} \in \phi^{-1}(y) \mid \xi_t = x)\mathbb{E}[\![\tau' \mid \xi_0' = y]\!] \geq \sum_{i=0}^{j_1} p(i) \cdot \mathbb{E}_{rls}(j_1 - i)$$

$$\geq \sum_{i=0}^{j} p(i) \cdot \mathbb{E}_{rls}(j - i) = n \sum_{i=0}^{j-1}(\prod_{p=1}^{\lambda}(\sum_{k=0}^{i}\binom{j_p}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{j_p-k})). \tag{4}$$

By comparing Eq. (3) with Eq. (4), we have $\forall x \notin \mathcal{X}^*$,

$$\sum_{y \in \mathcal{Y}} P(\xi_{t+1} \in \phi^{-1}(y) \mid \xi_t = x)\mathbb{E}[\![\tau'|\xi_0'=y]\!] - \sum_{y \in \mathcal{Y}} P(\xi_1' = y \mid \xi_0' = \phi(x))\mathbb{E}[\![\tau'|\xi_1'=y]\!]$$

$$\geq n(\sum_{i=0}^{j-1}(\prod_{p=1}^{\lambda}(\sum_{k=0}^{i}\binom{j_p}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{j_p-k})) - j) + 1$$

$$\geq n(\sum_{i=0}^{j-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda - j) + 1$$

$$\geq n(\sum_{i=0}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda - n) + 1,$$

where the 2nd '$\geq$' is because from Lemma 3, $\sum_{k=0}^{i}\binom{m}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{m-k}$ reaches the minimum when $m = n$, and the last '$\geq$' is by $(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda \leq 1$. When $x \in \mathcal{X}^*$, both Eq. (3) and Eq. (4) equal 0, because both chains are absorbing and the mapping ϕ is optimality-aligned. Thus, Eq. (1) in Theorem 1 holds with $\rho_t = (n(\sum_{i=0}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda - n) + 1)(1 - \pi_t(\mathcal{X}^*))$. By switch analysis,

$$\mathbb{E}[\![\tau|\xi_0 \sim \pi_0]\!] \geq \mathbb{E}[\![\tau'|\xi_0' \sim \pi_0^\phi]\!]$$

$$+ (n(\sum_{i=0}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda - n) + 1)\sum_{t=0}^{+\infty}(1 - \pi_t(\mathcal{X}^*)).$$

Since $\sum_{t=0}^{+\infty}(1 - \pi_t(\mathcal{X}^*)) = \mathbb{E}[\![\tau|\xi_0 \sim \pi_0]\!]$, we have

$$\mathbb{E}[\![\tau|\xi_0 \sim \pi_0]\!] \geq \frac{\mathbb{E}[\![\tau'|\xi_0' \sim \pi_0^\phi]\!]}{n(n - \sum_{i=0}^{n-1}(\sum_{k=0}^{i}\binom{n}{k}(\frac{1}{n})^k(1-\frac{1}{n})^{n-k})^\lambda)} \geq \frac{\mathbb{E}[\![\tau'|\xi_0' \sim \pi_0^\phi]\!]}{n\lceil\frac{e(c+1)\ln n}{\ln\ln n}\rceil}, \tag{5}$$

where the last inequality is by Lemma 4, since $\lambda \leq n^c$ for some constant c.

We then investigate $\mathbb{E}[\![\tau'|\xi_0' \sim \pi_0^\phi]\!]$. Since each of the μ solutions in the initial population is selected uniformly and randomly from $\{0,1\}^n$, we have

$$\forall 0 \leq j \leq n : \pi_0^\phi(\{y \in \mathcal{Y} \mid |y|_0 = j\}) = \pi_0(\{x \in \mathcal{X} \mid \min_{y \in x} |y|_0 = j\})$$

$$= \frac{(\sum_{k=j}^n \binom{n}{k})^\mu - (\sum_{k=j+1}^n \binom{n}{k})^\mu}{2^{n\mu}},$$

where $\sum_{k=j}^n \binom{n}{k}$ is the number of solutions with not less than j number of 0-bits. Then,

$$\mathbb{E}[\![\tau'|\xi_0' \sim \pi_0^\phi]\!] = \sum_{j=0}^n \pi_0^\phi(\{y \in \mathcal{Y} \mid |y|_0 = j\}) \mathbb{E}_{rls}(j)$$

$$= \frac{1}{2^{n\mu}} \sum_{j=1}^n ((\sum_{k=j}^n \binom{n}{k})^\mu - (\sum_{k=j+1}^n \binom{n}{k})^\mu) nj = \frac{n}{2^{n\mu}} \sum_{j=1}^n (\sum_{k=j}^n \binom{n}{k})^\mu$$

$$> n \sum_{j=1}^{\lfloor \frac{n}{4} \rfloor + 1} (\sum_{k=j}^n \binom{n}{k}/2^n)^\mu > \frac{n^2}{4} (\sum_{k=\lfloor \frac{n}{4} \rfloor + 1}^n \binom{n}{k}/2^n)^\mu = \frac{n^2}{4} (1 - \sum_{k=0}^{\lfloor \frac{n}{4} \rfloor} \binom{n}{k}/2^n)^\mu$$

$$\geq \frac{n^2}{4}(1 - 2^{H(\frac{1}{4})n - n})^\mu \geq \frac{n^2}{4} e^{-\frac{\mu}{2^{(1-H(\frac{1}{4}))n} - 1}} > \frac{n^2}{4} e^{-\frac{\mu}{1.13^n - 1}},$$

where the third inequality is by Lemma 5, the fourth inequality is by $\forall 0 < x < 1 :$ $(1-x)^y \geq e^{-\frac{xy}{1-x}}$, and the last inequality is by $2^{1-H(\frac{1}{4})} > 1.13$.

Applying the above lower bound on $\mathbb{E}[\![\tau'|\xi_0' \sim \pi_0^\phi]\!]$ to Eq. (5), we get, noting that μ is upper bounded by a polynomial in n,

$$\mathbb{E}[\![\tau|\xi_0 \sim \pi_0]\!] \geq \frac{n}{4\lceil \frac{e(c+1)\ln n}{\ln \ln n} \rceil} e^{-\frac{\mu}{1.13^n - 1}}, \quad \text{i.e.,} \quad \Omega(\frac{n \ln \ln n}{\ln n}).$$

Considering the μ number of fitness evaluations for the initial population and the λ number of fitness evaluations in each generation, the expected running time of the $(\mu+\lambda)$-EA on UBoolean is lower bounded by $\Omega(\mu + \frac{\lambda n \ln \ln n}{\ln n})$. Because the $(\mu+\lambda)$-EA belongs to mutation-based EAs, we can also directly use the general lower bound $\Omega(n \ln n)$ [18]. Thus, the theorem holds. $\qquad\square$

5 Conclusion

This paper analyzes the expected running time of the $(\mu+\lambda)$-EA for solving a general problem class consisting of pseudo-Boolean functions with a unique global optimum. We derive the lower bound $\Omega(n \ln n + \mu + \lambda n \ln \ln n / \ln n)$ by applying the recently proposed approach switch analysis. The results partially complete the running time comparison between the $(\mu+\lambda)$-EA and the (1+1)-EA on the two well-studied pseudo-Boolean problems, OneMax and LeadingOnes. We can now conclude that when μ or λ is slightly large, the $(\mu+\lambda)$-EA has a worse expected running time. The investigated $(\mu+\lambda)$-EA only uses mutation, while crossover is a characterizing feature of EAs. Therefore, we will try to analyze the running time of population-based EAs with crossover operators in the future.

References

1. Auger, A., Doerr, B.: Theory of Randomized Search Heuristics: Foundations and Recent Developments. World Scientific, Singapore (2011)
2. Bäck, T.: Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms. Oxford University Press, Oxford, UK (1996)
3. Chen, T., He, J., Chen, G., Yao, X.: Choosing selection pressure for wide-gap problems. Theoretical Computer Science 411(6), 926–934 (2010)
4. Chen, T., He, J., Sun, G., Chen, G., Yao, X.: A new approach for analyzing average time complexity of population-based evolutionary algorithms on unimodal problems. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 39(5), 1092–1106 (2009)
5. Chen, T., Tang, K., Chen, G., Yao, X.: A large population size can be unhelpful in evolutionary algorithms. Theoretical Computer Science 436(8), 54–70 (2012)
6. Doerr, B., Künnemann, M.: Optimizing linear functions with the $(1+\lambda)$ evolutionary algorithm - different asymptotic runtimes for different instances. Theoretical Computer Science 561, 3–23 (2015)
7. Droste, S., Jansen, T., Wegener, I.: On the analysis of the (1+1) evolutionary algorithm. Theoretical Computer Science 276(1-2), 51–81 (2002)
8. Flum, J., Grohe, M.: Parameterized Complexity Theory. Springer, New York, NY (2006)
9. Freĭdlin, M.: Markov Processes and Differential Equations: Asymptotic Problems. Birkhäuser Verlag, Basel, Switzerland (1996)
10. Gießen, C., Witt, C.: Population size vs. mutation strength for the $(1+\lambda)$ EA on OneMax. In: Proceedings of GECCO'15. pp. 1439–1446. Madrid, Spain (2015)
11. He, J., Yao, X.: Drift analysis and average time complexity of evolutionary algorithms. Artificial Intelligence 127(1), 57–85 (2001)
12. He, J., Yao, X.: From an individual to a population: An analysis of the first hitting time of population-based evolutionary algorithms. IEEE Transactions on Evolutionary Computation 6(5), 495–511 (2002)
13. Jansen, T., Wegener, I.: On the utility of populations in evolutionary algorithms. In: Proceedings of GECCO'01. pp. 1034–1041. San Francisco, CA (2001)
14. Lehre, P.K., Witt, C.: Black-box search by unbiased variation. Algorithmica 64(4), 623–642 (2012)
15. Lehre, P.K., Yao, X.: On the impact of mutation-selection balance on the runtime of evolutionary algorithms. IEEE Transactions on Evolutionary Computation 16(2), 225–241 (2012)
16. Neumann, F., Witt, C.: Bioinspired Computation in Combinatorial Optimization: Algorithms and Their Computational Complexity. Springer-Verlag, Berlin, Germany (2010)
17. Storch, T.: On the choice of the parent population size. Evolutionary Computation 16(4), 557–578 (2008)
18. Sudholt, D.: A new method for lower bounds on the running time of evolutionary algorithms. IEEE Transactions on Evolutionary Computation 17(3), 418–435 (2013)
19. Witt, C.: Runtime analysis of the $(\mu+1)$ EA on simple pseudo-Boolean functions. Evolutionary Computation 14(1), 65–86 (2006)
20. Witt, C.: Population size versus runtime of a simple evolutionary algorithm. Theoretical Computer Science 403(1), 104–120 (2008)
21. Yu, Y., Qian, C., Zhou, Z.H.: Switch analysis for running time analysis of evolutionary algorithms. IEEE Transactions on Evolutionary Computation 19(6), 777–792 (2015)
22. 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)
23. Yu, Y., Qian, C.: Running time analysis: Convergence-based analysis reduces to switch analysis. In: Proceedings of CEC'15. pp. 2603–2610. Sendai, Japan (2015)