Drift Analysis with Fitness Levels for Elitist Evolutionary Algorithms

Jun He jun.he@ntu.ac.uk
Department of Computer Science, Nottingham Trent University, Nottingham NG11 8NS, United Kingdom

Yuren Zhou zhouyuren@mail.sysu.edu.cn
School of Data and Computer Science, Sun Yatsen University, Guangzhou, 510006, China

Abstract

The fitness level method is a popular tool for analyzing the hitting time of elitist evolutionary algorithms. Its idea is to divide the search space into multiple fitness levels and estimate lower and upper bounds on the hitting time using transition probabilities between fitness levels. However, the lower bound generated by this method is often loose. An open question regarding the fitness level method is what are the tightest lower and upper time bounds that can be constructed based on transition probabilities between fitness levels. To answer this question, we combine drift analysis with fitness levels and define the tightest bound problem as a constrained multi-objective optimization problem subject to fitness levels. The tightest metric bounds by fitness levels are constructed and proven for the first time. Then linear bounds are derived from metric bounds and a framework is established that can be used to develop different fitness level methods for different types of linear bounds. The framework is generic and promising, as it can be used to draw tight time bounds on both fitness landscapes with and without shortcuts. This is demonstrated in the example of the (1+1) EA maximizing the TwoMax1 function.

Keywords

Evolutionary algorithm; algorithm analysis; hitting time; fitness levels; drift analysis; Markov chain

1 Introduction

1.1 Background

The time complexity of evolutionary algorithms (EAs) is an important topic in the EA theory (Oliveto et al., 2007; Yu and Zhou, 2008; Doerr and Neumann, 2019; Huang et al., 2019). The computation time of EAs can be measured by either the number of generations to find an optimum for the first time, called hitting time (He and Yao, 2001), or the number of fitness evaluations, called running time (He and Yao, 2017). The analysis of running time is more complicated as it is related to the population size (He and Yao, 2002; Chen et al., 2009; He and Yao, 2017), and the population size often varies from generation to generation. Therefore, we limit our discussion to hitting time. Several methods have been proposed for analyzing the hitting time of EAs, such as drift analysis (He and Yao, 2001), Markov chains (He and Yao, 2002, 2003) and fitness level partition (Wegener, 2001). Each method has its own advantages and disadvantages. Drift analysis is a powerful tool in which an appropriate distance is constructed as a bound on hitting time (He and Yao, 2001; Oliveto and Witt, 2011; Doerr et al., 2012). According to the theory of absorbing Markov chains, the exact hitting time of EAs can be calculated from the fundamental matrix of absorbing Markov chains (He and Yao, 2002, 2003; Zhou et al., 2009).

The fitness level method (Wegener, 2001, 2003) is a popular tool used to estimate the hitting time of elitist EAs (Antipov et al., 2018; Corus et al., 2020; Rajabi and Witt, 2020; Quinzan et al., 2021; Aboutaib and Sutton, 2022; Malalanirainy and Moraglio, 2022; Oliveto et al., 2022). The basic concept of this method is to divide the search space into multiple ranks $(S_0, \cdots, S_K)$, called fitness levels, based on the fitness value from high to low, where the highest rank S_0 is the optimal set; then calculate transition probabilities between fitness levels; finally, estimate a bound d_k on the hitting time of the EA starting from S_k. The method was combined with other techniques such as tail bounds (Witt, 2014) and stochastic domination (Doerr, 2019). The fitness level method is available for elitist EAs. Although the level partition is also used to analyze non-elitist EAs (Corus et al., 2017; Case and Lehre, 2020), they should be considered as a different method.

In this paper, we express time bounds by fitness levels in linear forms:

$$\text{lower bound } d_k = \sum_{\ell=1}^{k} \frac{c_{k,\ell}}{\max_{X_\ell \in S_\ell} p(X_\ell, S_0 \cup \cdots \cup S_{\ell-1})}, \tag{1}$$

$$\text{upper bound } d_k = \sum_{\ell=1}^{k} \frac{c_{k,\ell}}{\min_{X_\ell \in S_\ell} p(X_\ell, S_0 \cup \cdots \cup S_{\ell-1})}, \tag{2}$$

where $c_{k,\ell}$ are coefficients and $p(X_\ell, S_0 \cup \cdots \cup S_{\ell-1})$ represents the transition probability from $X_\ell \in S_\ell$ to $S_0 \cup \cdots \cup S_{\ell-1}$.

How to calculate coefficients $c_{k,\ell}$ for tight bounds is the key topic in the fitness level method. Wegener (2003) assigned $c_{k,k} = 1$, other coefficients $c_{k,\ell} = 0$ for the lower bound and $c_{k,\ell} = 1$ for the upper bound where $k > \ell$. This assignment is good at obtaining a tight upper bound, but not good at obtaining a tight lower bound. Several efforts have been made to improve the lower bound since then. Sudholt (2012) made an improvement using a constant coefficient $c_{k,\ell} = c$ (called viscosity) for $k > \ell$ and $c_{k,k} = 1$, and gave tight lower time bounds of the (1+1) EA on several unimodal functions such as LeadingOnes, OneMax, long k-paths. Recently, Doerr and Kötzing (2022) made another improvement using coefficients $c_{k,\ell} = c_\ell$ (called visit probability) and provided tight lower bounds of the (1+1) EA on LeadingOnes, OneMax, long k-paths jump function. However, in this paper we find that the lower bounds based on c or c_ℓ are loose on landscapes with shortcuts. A shortcut means that some intermediate fitness level is skipped with a large probability. Therefore, it is necessary to improve the lower bound of EAs on fitness landscapes with shortcuts.

1.2 New research and main results in this paper

The aim of this paper is to explore two research questions that have not been addressed before.

1. What are the tightest lower and upper bounds that can be constructed using transition probabilities between fitness levels?

2. Is it possible to use fitness level methods to draw tight lower bounds on fitness landscapes with shortcuts?

To answer the questions, we combine drift analysis with fitness levels for constructing lower and upper bounds on the hitting time of elitist EAs. The fitness level method is viewed as a combination of drift analysis and fitness levels. Given a fitness level partition $(S_0, \cdots, S_K)$, a distance d_k between S_k and S_0 is constructed using transition probabilities between fitness levels. This distance d_k is called a metric bound. Then by drift analysis, it is proved that d_k is a lower or upper bound on the hitting time of the EA starting from S_k, and the best d_k^* is the tightest metric bound. The new contributions and results are summarized in three parts.

1. First, we construct metric bounds by fitness levels and prove that the best metric bounds are the tightest. The metric lower bound is expressed recursively as

$$d_k \leq \min_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_0 \cup \cdots \cup S_{k-1})} + \sum_{\ell=1}^{k-1} \frac{p(X_k, S_\ell)}{p(X_\ell, S_0 \cup \cdots \cup S_{\ell-1})} d_\ell \right\}. \tag{3}$$

   Similarly, the metric upper bound is expressed recursively as

$$d_k \geq \max_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_0 \cup \cdots \cup S_{k-1})} + \sum_{\ell=1}^{k-1} \frac{p(X_k, S_\ell)}{p(X_\ell, S_0 \cup \cdots \cup S_{\ell-1})} d_\ell \right\}. \tag{4}$$

   We call the above bounds Type-$r_{k,\ell}$ bounds. The tightest lower or upper bound is reached when Inequality (3) or (4) is an equality.

2. Secondly, we construct general linear bounds from metric bounds (3) and (4). The lower bound coefficients in (1) satisfy $c_{k,k} = 1$ and linear inequalities:

$$c_{k,\ell} \leq \min_{X_k \in S_k} \frac{p(X_k, S_\ell) + \sum_{j=\ell+1}^{k-1} p(X_k, S_j) c_{j,\ell}}{p(X_k, S_0 \cup \cdots \cup S_{k-1})}, \qquad 0 < \ell < k. \tag{5}$$

   The upper bound coefficients in (2) satisfy $c_{k,k} = 1$ and linear inequalities:

$$c_{k,\ell} \geq \max_{X_k \in S_k} \frac{p(X_k, S_\ell) + \sum_{j=\ell+1}^{k-1} p(X_k, S_j) c_{j,\ell}}{p(X_k, S_0 \cup \cdots \cup S_{k-1})}, \qquad 0 < \ell < k. \tag{6}$$

   We call the above bounds Type-$c_{k,\ell}$ bounds. Previous bounds (Wegener, 2003; Sudholt, 2012; Doerr and Kötzing, 2022) can be regarded as special cases of $c_{k,\ell} = 0, 1, c, c_\ell$, named Type-$0, 1, c, c_\ell$ bounds respectively.

3. Finally, we demonstrate the advantage of the Type-$c_{k,\ell}$ lower bound over Type-c and c_ℓ lower bounds. For the (1+1) EA maximizing the TwoMax1 function, we prove that theType-$c_{k,\ell}$ lower bound is $\Omega(n \ln n)$, but Type-c and c_ℓ lower bounds are only $O(1)$.

The paper is organized as follows: Section 2 provides the foundation of theoretical analysis. Section 3 reviews existing fitness level methods and explains the necessity of improving previous lower bounds. Section 4 proposes drift analysis with fitness levels, constructs new metric bounds and proves they are the tightest. Section 5 constructs general linear bounds and presents different explicit expressions of coefficients. Section 6 shows the application of general linear bounds. Section 7 concludes the work.

2 Preliminary

2.1 Elitist EAs and Markov chains

A maximization problem is considered in the paper: $f_{\max} = \max f(x)$ where $f(x)$ is defined on a finite set. In EAs, an individual x represents a solution. A population consists of several individuals, denoted by X. The fitness of a population $f(X) = \max\{f(x); x \in X\}$. Let S denote the set of all populations and S_{opt} the set of optimal populations X_{opt} such that $f(X_{\text{opt}}) = f_{\max}$. This paper studies elitist EAs that maximize $f(x)$. Let $X^{[t]}$ denote the t-th generation population. An EA is elitist if $f(X^{[t]}) \geq f(X^{[t-1]})$.

A simple elitist EA is the (1+1) EA using bitwise mutation and elitist selection for maximizing a pseudo-Boolean function: $f(x)$ where $x = (x_1, \cdots, x_n) \in \{0,1\}^n$. The (1+1) EA does not use a population, but keeps only an individual.

Algorithm 1 The (1+1) EA that maximizes a pseudo-Boolean function $f(x)$

```
initialize a solution x and let x^[0] = x;
for t = 1, 2, ··· do
    flip each bit of x independently with probability 1/n and generate a solution y;
    if f(y) ≥ f(x), then let x^[t] = y, otherwise x^[t] = x.
end for
```

We assume that EAs are modeled by homogeneous Markov chains (He and Yao, 2003; He and Lin, 2016). The set S is the state space of a Markov chain and a population X is a state. The Markov chain property means that the next state depends only on the current state, that is, $\Pr(X^{[t+1]} \mid X^{[t]}, \ldots, X^{[0]}) = \Pr(X^{[t+1]} \mid X^{[t]})$. The homogeneous property means that the transition probability from a state to another does not change over generations, that is, $\Pr(X^{[t+1]} = Y \mid X^{[t]} = X) = p(X, Y)$.

2.2 Hitting time and drift analysis

Hitting time is the first time when an EA finds an optimal solution.

Definition 1. *Given an elitist EA for maximizing $f(x)$, assume that the initial population $X^{[0]} = X$.* Hitting time $\tau(X)$ *is the number of generations when an optimum is found for the first time, that is, $\tau(X) = \min\{t \geq 0, X^{[t]} \in S_{\text{opt}} \mid X^{[0]} = X\}$. The* mean hitting time *$m(X)$ is the expected value: $m(X) = \mathrm{E}[\tau(X)]$. Assume that $X^{[0]}$ is chosen at random, the* mean hitting time *m is the expected value: $m = \mathrm{E}[m(X^{[0]})]$.*

Drift analysis was introduced by He and Yao (2001) to the analysis of hitting time of EAs. It is based on the intuitive idea: time = distance/drift. A non-negative function $d(X)$ measures the distance between X and the optimal set. By default, let $d(X) = 0$ if $X \in S_{\text{opt}}$. A distance function $d(X)$ is called a lower time bound if for all X, $d(X) \leq m(X)$, while $d(X)$ is called an upper time bound if for all X, $d(X) \geq m(X)$.

There are several variants of drift analysis (He and Yao, 2001; Oliveto and Witt, 2011; Doerr et al., 2012; Doerr and Goldberg, 2013). For a complete review of drift analysis, see (Kötzing and Krejca, 2019; Lengler, 2020). In the current paper, we establish drift analysis with fitness levels based on the Markov chain version of drift analysis (He and Yao, 2003). For elitist EAs that cannot be modeled by Markov chains, it is still possible to establish drift analysis with fitness levels by the super-martingale version of drift analysis (He and Yao, 2001).

Definition 2. *The* drift $\Delta d(X)$ *is the distance change per generation,*

$$\Delta d(X) = d(X) - \sum_{Y \in S} p(X,Y)d(Y). \tag{7}$$

Lemma 1. *(He and Yao, 2003, Theorem 3) If for any $X \notin S_{\text{opt}}$, the drift $\Delta d(X) \leq 1$, then the mean hitting time $m(X) \geq d(X)$.*

Lemma 2. *(He and Yao, 2003, Theorem 2) If for any $X \notin S_{\text{opt}}$, the drift $\Delta d(X) \geq 1$, then the mean hitting time $m(X) \leq d(X)$.*

We use the asymptotic notation (Knuth, 1976) to describe the tightness of lower and upper bounds. The worst-case time complexity of an EA is measured by the maximum value of the mean hitting time $\max_{X \in S} m(X)$. We say that a lower bound $d(X)$ is tight if $\max_{X \in S} d(X) = \Omega(\max_{X \in S} m(X))$, and an upper bound $d(X)$ is tight if $\max_{X \in S} d(X) = O(\max_{X \in S} m(X))$. We also divide coefficients $c_{k,\ell}$ into two categories: large coefficients $c_{k,\ell}$ if $c_{k,\ell} = \Omega(1)$ and small coefficients $c_{k,\ell}$ if $c_{k,\ell} = o(1)$.

2.3 Fitness level partition and transition probabilities between fitness levels

The fitness level method depends on a fitness level partition.

Definition 3. *In a* fitness level partition, *the state space S is divided into $K+1$ disjoint subsets (ranks) $(S_0, \cdots, S_K)$ according to the fitness from high to low such that (i) the highest rank $S_0 = S_{\text{opt}}$, (ii) for all $X_k \in S_k$ and $X_{k+1} \in S_{k+1}$, the rank order holds: $f(X_k) > f(X_{k+1})$. Each rank is called a* fitness level.

The fitness level method is based on transition probabilities between fitness levels. The notation $p(X_k, S_\ell)$ denotes the transition probability from $X_k \in S_k$ to S_ℓ.

$$p(X_k, S_\ell) = \Pr(X^{[t+1]} \in S_\ell \mid X^{[t]} = X_k). \tag{8}$$

Its minimal and maximal values are denoted by

$$p_{\min}(X_k, S_{[0,k-1]}) = \min_{X_k \in S_k} p(X_k, S_{[0,k-1]}),$$
$$p_{\max}(X_k, S_{[0,k-1]}) = \max_{X_k \in S_k} p(X_k, S_{[0,k-1]}).$$

Other transition probabilities between levels are derived from $p(X_k, S_\ell)$. Let $[i,j]$ denote the index set $\{i, i+1, \cdots, j-1, j\}$ and $S_{[i,j]}$ denote the union of levels $S_i \cup S_{i+1} \cup \cdots \cup S_{j-1} \cup S_j$. The transition probability from $X_k \in S_k$ to $S_{[i,j]}$ is denoted by $p(X_k, S_{[i,j]})$.

The notation $r(X_k, S_\ell)$ denotes the conditional probability

$$r(X_k, S_\ell) = \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,k-1]})}. \tag{9}$$

Its minimal and maximal values are denoted by

$$r_{\min}(X_k, S_\ell) = \min_{X_k \in S_k} p(X_k, S_\ell),$$
$$r_{\max}(X_k, S_\ell) = \max_{X_k \in S_k} r(X_k, S_\ell).$$

Table 1 lists main symbols used in this paper.

Table 1: Notation used in the paper.

S_k	a fitness level
$S_{[i,j]}$	the union of fitness levels $S_i \cup S_{i+1} \cdots \cup S_{j-1} \cup S_j$ where $i < j$
X_k	a state in S_k
$m(X_k)$	the mean hitting time when the EA starts from X_k
$p(X_k, S_\ell)$	the transition probability from X_k to S_ℓ
$p(X_k, S_{[i,j]})$	the transition probability from X_k to $S_{[i,j]}$
$r(X_k, S_\ell)$	the conditional probability $\frac{p(X_k,S_\ell)}{p(X_k,S_{[0,k-1]})}$
$c_{k,\ell}, c_\ell, c$	coefficients in linear bounds

2.4 Shortcuts

Intuitively, the behavior of an elitist EA searching for the maximum value of a fitness function can be viewed as climbing on a fitness landscape. For most fitness landscapes, an EA can take different paths from lower to higher fitness levels, some of which are shorter than others. A shortcut implies that an intermediate fitness level is skipped. In this paper, we formally define shortcuts as follows.

Definition 4. *Given an elitist EA for maximizing a function $f(x)$ and a fitness level partition $(S_0, \cdots, S_K)$, there exists a* shortcut *from X_k to $S_{[0,\ell-1]}$ skipping S_ℓ (where $1 \le \ell < k$) if the conditional probability*

$$\frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})} = o(1). \tag{10}$$

According to (10), the conditional probability of the EA starting from X_k to visit S_ℓ is $o(1)$. Thus, S_ℓ is skipped with a large conditional probability $1 - o(1)$.

Fitness landscapes can be divided into two categories: with shortcuts and without shortcuts. Let us demonstrate two examples. The first example is the (1+1) EA that maximizes the OneMax function:

$$\mathrm{OM}(x) = |x|, \quad x = (x_1, \cdots, x_n) \in \{0,1\}^n,$$

where $|x| = x_1 + \cdots + x_n$. The state space can be divided into $n+1$ levels $(S_0, \cdots, S_n)$, where $S_k = \{x \in \{0,1\}^n; \mathrm{OM}(x) = n - k\}$. Figure 1 shows that no shortcut exists on the fitness landscape of the (1+1) EA on OneMax.

The second example is the (1+1) EA maximizing the TwoMax1 function.

$$\mathrm{TM1}(x) = \begin{cases} n & \text{if } |x| = 0 \text{ or } |x| = n, \\ |x| & \text{if } |x| \ge \frac{n}{2}, \\ \frac{n}{2} - |x| & \text{else,} \end{cases}$$

where n is a large even integer. There are two maxima at $|x| = 0$ and n. TwoMax1 is a variant of the TwoMax function defined in (He et al., 2015). The search space can be split into n fitness levels $(S_0, \cdots, S_{n-1})$ from high to low: $S_k = \{x \in \{0,1\}^n : \mathrm{TM1}(x) = n - k\}$. Figure 1 displays two shortcuts on the fitness landscape of the (1+1) EA on TwoMax1. The two solid lines represent shortcuts. The first shortcut is $S_{n/2+1} \to S_0$ skipping $S_1, \cdots, S_{n/2}$. The second shortcut is $S_{n-1} \to S_{n/2}$ skipping $S_{n/2+1}$. We omit the rigorous proof of these shortcuts.

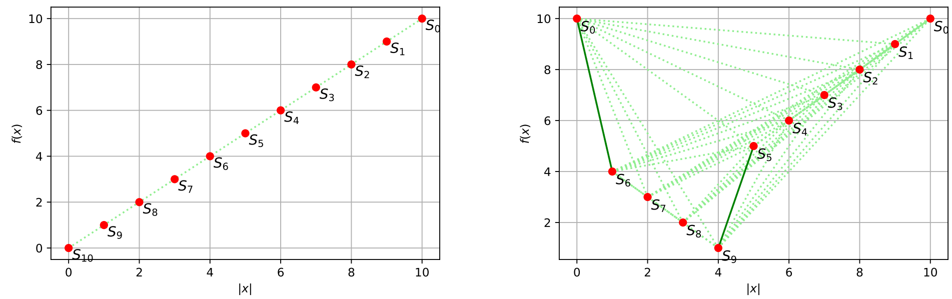

Figure 1: Left: the (1+1) EA on OneMax(x) where $n = 10$. Right: The (1+1) EA on TwoMax1(x) where $n = 10$. Dotted lines represent transitions. Solid lines are two shortcuts: $S_6 \to S_0$ skipping $S_1, \cdots, S_5$, and $S_9 \to S_5$ skipping S_6.

3 Review and discussion of existing fitness level methods

3.1 Existing fitness level methods

Given a fitness level partition $(S_0, \cdots S_K)$, previous results are summarized as follows. Wegener (2003) gave simple Type-0 lower bound and Type-1 upper bound.

Proposition 1. *(Wegener, 2003, Lemma 1) For all $k \geq 1$ and $X_k \in S_k$, the mean hitting time $m(X_k) \geq \frac{1}{p_{\max}(X_k, S_{[0,k-1]})}$.*

Proposition 2. *(Wegener, 2003, Lemma 2) For all $k \geq 1$ and $X_k \in S_k$, the mean hitting time $m(X_k) \leq \sum_{\ell=1}^{k} \frac{1}{p_{\min}(X_\ell, S_{[0,\ell-1]})}$.*

Sudholt (2012) improved the lower bound using a constant coefficient c (called viscosity).

Proposition 3. *(Sudholt, 2012, Theorem 3) For any $0 \leq \ell < k \leq K$, let $p(X_k, S_\ell) \leq p_{\max}(X_k, S_{[0,k-1]})\, r_{k,\ell}$ and $\sum_{\ell=0}^{k-1} r_{k,\ell} = 1$. Assume that there is some $0 \leq c \leq 1$ such that for any $1 \leq l < k \leq K$, it holds $r_{k,\ell} \geq c \sum_{j=0}^{\ell} r_{k,j}$. Then the mean hitting time*

$$m(X^{[0]}) \geq \sum_{k=1}^{K} \Pr(X^{[0]} \in S_k) \left(\frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c}{p_{\max}(X_\ell, S_{[0,\ell-1]})} \right).$$

An issue with Proposition 3 is that the above Type-c lower bound is loose on fitness landscapes with shortcuts. This issue is demonstrated by an example in Section 3.2.

Sudholt (2012) also gave an upper bound similar to Proposition 3 but with an extra condition $(1-c)p_{\min}(X_{\ell+1}, S_{[0,\ell]}) \leq p_{\min}(X_\ell, S_{[0,\ell-1]})$.

Proposition 4. *(Sudholt, 2012, Theorem 4) For any $0 \leq \ell < k \leq K$, let $p(X_k, S_\ell) \geq p_{\min}(X_k, S_{[0,k-1]})\, r_{k,\ell}$ and $\sum_{\ell=0}^{k-1} r_{k,\ell} = 1$. Assume that there is some $0 \leq c \leq 1$ such that for all $1 \leq l < k \leq K$, it holds $r_{k,\ell} \leq c \sum_{j=0}^{\ell} r_{k,j}$. Further, assume that for all $1 \leq l \leq K-2$, it holds $(1-c)p_{\min}(X_{\ell+1}, S_{[0,\ell]}) \leq p_{\min}(X_\ell, S_{[0,\ell-1]})$. Then the mean hitting time*

$$m(X^{[0]}) \leq \sum_{k=1}^{K} \Pr(x^{[0]} \in S_k) \left(\frac{1}{p_{\min}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c}{p_{\min}(X_\ell, S_{[0,\ell-1]})} \right).$$

Doerr and Kötzing (2022) further improved the lower bound using coefficients c_ℓ (called visit probability).

Proposition 5. *(Doerr and Kötzing, 2022, Theorem 8) For all $\ell = 1, \cdots, K$, let c_ℓ be a lower bound on the probability of there being a t such that $X^{[t]} \in S_\ell$. Then the mean hitting time $m(X^{[0]}) \geq \sum_{\ell=1}^{K} \frac{c_\ell}{p_{\max}(X_\ell, S_{[0,\ell-1]})}$.*

Proposition 5 does not provide an explicit formula to calculate c_ℓ. Therefore, Doerr and Kötzing (2022) proposed a method of calculating c_ℓ as follows.

Lemma 3. *(Doerr and Kötzing, 2022, Lemma 10) For $1 \leq \ell \leq K$, suppose there is c_ℓ such that two conditions*

$$c_\ell \leq \min\{\frac{p(X, S_\ell)}{p(X, S_{[0,\ell]})} : X \in S_{[\ell+1,K]}, p(X, S_{[0,\ell]}) > 0\}, \tag{11}$$

$$c_\ell \leq \frac{\Pr(X^{[0]} \in S_\ell)}{\Pr(X^{[0]} \in S_{[0,\ell]})}. \tag{12}$$

Then c_ℓ is a lower bound for visiting S_ℓ as required by Proposition 5.

An issue with Lemma 3 is that the above Type-c_ℓ lower bound (11) is loose on fitness landscapes with shortcuts. An example in Section 3.3 shows this issue.

Doerr and Kötzing (2022) also gave an upper bound similar to Proposition 5. But they do not provide an explicit formula to calculate c_ℓ using transition probabilities between fitness levels.

Proposition 6. *(Doerr and Kötzing, 2022, Theorem 9) For all $\ell = 1, \cdots, K$, let c_ℓ be an upper bound on the probability of there being a t such that $X^{[t]} \in S_\ell$. Then the mean hitting time $m(X^{[0]}) \leq \sum_{\ell=1}^{K} \frac{c_\ell}{p_{\min}(X_\ell, S_{[0,\ell-1]})}$.*

3.2 Case Study 1: A loose Type-c lower bound for the (1+1) EA on TwoMax1

In this case study, we find that the Type-c lower bound by Proposition 3 is loose on fitness landscapes with shortcuts. Consider the (1+1) EA on TwoMax1. Assume that the EA starts from S_{n-1}. We prove that the lower bound by Proposition 3,

$$d_{n-1} = \frac{1}{p(x_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c}{p(x_\ell, S_{[0,\ell-1]})}, \tag{13}$$

is only $O(1)$.

Transition probabilities $p(x_\ell, S_{[0,\ell-1]})$ (where $\ell = 1, \cdots, n/2$) are calculated as follows. Since x_ℓ has ℓ zero-valued bits. The transition from x_ℓ to $S_{\ell-1} \subset S_{[0,\ell-1]}$ happens if 1 zero-valued bit is flipped and other bits are unchanged. Thus

$$p(x_\ell, S_{[0,\ell-1]}) \geq \binom{\ell}{1} \frac{1}{n} \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{\ell}{n} e^{-1}. \tag{14}$$

Transition probabilities $p(x_\ell, S_{[0,\ell-1]})$ (where $\ell = n/2+1, \cdots, n-1$) are calculated as follows. Since x_ℓ has $\ell - n/2$ one-valued bits, the transition from x_ℓ to $S_{[0,\ell-1]}$ happens if 1 one-valued bit is flipped and other bits are unchanged. Thus

$$p(x_\ell, S_{[0,\ell-1]}) \geq \binom{\ell - n/2}{1} \frac{1}{n} \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{\ell - n/2}{n} e^{-1}. \tag{15}$$

Inserting (14) and (15) into (13), we get

$$d_{n-1} \leq \frac{en}{(n-1)-\frac{n}{2}} + c\left(\sum_{\ell=1}^{n/2} \frac{en}{\ell} + \sum_{\ell=n/2+1}^{n-2} \frac{en}{\ell - n/2}\right). \tag{16}$$

Coefficient c is calculated using the shortcut $S_{n/2+1} \to S_0$ as follows. We replace $r_{k,\ell}$ in Proposition 3 with $r(x_k, S_\ell)$. Let $k = n/2+1$ and $\ell = 1$. According to Proposition 3,

$$c \leq \frac{r_{n/2+1,1}}{r_{n/2+1,0} + r_{n/2+1,1}} = \frac{p(x_{n/2+1}, S_1)}{p(x_{n/2+1}, S_0) + p(x_{n/2+1}, S_1)}. \tag{17}$$

Since $x_{n/2+1}$ has $n-1$ zero-valued bits and 1 one-valued bit, the transition from $x_{n/2+1}$ to S_0 happens if the one-valued bit is flipped and other bits are unchanged.

$$p(x_{n/2+1}, S_0) \geq \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}. \tag{18}$$

The transition from $x_{n/2+1}$ to S_1 happens if and only if $n-2$ zero-valued bits are flipped and other bits are unchanged.

$$p(x_{n/2+1}, S_1) = \left(\frac{1}{n}\right)^{n-2}\left(1 - \frac{1}{n}\right)^{2}. \tag{19}$$

Inserting (18) and (19) into (17), we have

$$c \leq O(n^{-n+3}). \tag{20}$$

Inserting (20) into (16), we get

$$d_{n-1} \leq \frac{en}{(n-1)-\frac{n}{2}} + O(n^{-n+3})\left(\sum_{\ell=1}^{n/2} \frac{en}{\ell} + \sum_{\ell=n/2+1}^{n-2} \frac{en}{\ell - n/2}\right) = O(1).$$

The lower bound d_{n-1} is $O(1)$, much looser than the actual hitting time $\Theta(n \ln n)$.

3.3 Case Study 2: A loose Type-c_ℓ lower bound for the (1+1) EA on TwoMax1

In this case study, we find that the Type-c_ℓ lower bound by Lemma 3 and Proposition 5 is loose on fitness landscapes with shortcuts. Consider the (1+1) EA on TwoMax1 under random initialization. We prove that the lower bound by Lemma 3 and Proposition 5,

$$d = \sum_{\ell=1}^{n-1} \frac{c_\ell}{p_{\max}(x_\ell, S_{[0,\ell-1]})}, \tag{21}$$

is only $O(1)$.

Inserting (14) and (15) into (21), we get

$$d \leq c_1 \frac{en}{(n-1)-\frac{n}{2}} + \sum_{\ell=1}^{n/2} c_\ell \frac{en}{\ell} + \sum_{\ell=n/2+1}^{n-2} c_\ell \frac{en}{\ell - n/2}. \tag{22}$$

Coefficients c_ℓ are calculated by Condition (11) but without Condition (12), because Condition (12) only reduces the c_ℓ value and makes the lower bound smaller.

For $\ell = 1, \cdots, n/2$, coefficients c_ℓ are calculated using the shortcut $S_{n/2+1} \to S_0$ as follows. According to Condition (11), we have

$$c_\ell \le \frac{p(x_{n/2+1}, S_\ell)}{p(x_{n/2+1}, S_{[0,\ell]})}. \tag{23}$$

Since $x_{n/2+1}$ has 1 one-valued bit and a state in S_ℓ has $n-\ell$ one-valued bits, the transition from $x_{n/2+1}$ to S_ℓ happens only if $n-1-\ell$ zero-valued bits are flipped. Thus,

$$p(x_{n/2+1}, S_\ell) \le \binom{n-1}{n-1-\ell} \left(\frac{1}{n}\right)^{n-1-\ell}. \tag{24}$$

The transition from $x_{n/2+1}$ to $S_0 \subset S_{[0,\ell]}$ happens if the one-valued bit is flipped and other bits are unchanged. Thus,

$$p(x_{n/2+1}, S_{[0,\ell]}) \ge \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1} \ge \frac{1}{en}. \tag{25}$$

Inserting (24) and (25) into (23), we have

$$c_\ell \le e\binom{n-1}{n-1-\ell} \left(\frac{1}{n}\right)^{n-\ell}, \quad \ell = 1, \cdots, n/2. \tag{26}$$

For $\ell = n/2+1, \cdots, n-2$, coefficients c_ℓ are calculated by the shortcut $S_{n-1} \to S_{n/2}$ as follows. According to Condition (11), we have

$$c_\ell \le \frac{p(x_{n-1}, S_\ell)}{p(x_{n-1}, S_{[0,\ell]})}, \quad \ell = n/2+1, \cdots, n-2. \tag{27}$$

Since x_{n-1} has $n/2-1$ one-valued bits and $n/2+1$ zero-valued bits, the transition from x_{n-1} to S_ℓ happens only if $n-1-\ell$ zero-valued bits are flipped. Thus,

$$p(x_{n-1}, S_\ell) \le \binom{n/2-1}{n-1-\ell} \left(\frac{1}{n}\right)^{n-1-\ell}. \tag{28}$$

The transition from x_{n-1} to $S_{[0,\ell]}$ happens if 1 zero-valued bit is flipped and other $n/2-1$ one-valued bits are unchanged. Thus,

$$p(x_{n-1}, S_{[0,\ell]}) \ge \binom{n/2+1}{1}\frac{1}{n}\left(1-\frac{1}{n}\right)^{n/2-1} \ge \frac{1}{2e}. \tag{29}$$

Inserting (28) and (29) into (27), we have

$$c_\ell \le 2e\binom{n/2-1}{n-1-\ell} \left(\frac{1}{n}\right)^{n-1-\ell}, \quad \ell = n/2+1, \cdots, n-2. \tag{30}$$

For $\ell = n-1$, we use the trivial estimation $c_{n-1} \le 1$.

Inserting (26) and (30) into (22), we get

$$
\begin{aligned}
d \leq & O(1) + \sum_{\ell=1}^{n/2} e \binom{n-1}{n-1-\ell} \left(\frac{1}{n}\right)^{n-\ell} \frac{en}{\ell} + \sum_{\ell=n/2+1}^{n-2} 2e \binom{n/2-1}{n-1-\ell} \left(\frac{1}{n}\right)^{n-1-\ell} \frac{en}{(\ell-n/2)} \\
\leq & O(1) + e^2 \sum_{\ell=1}^{n/2} \frac{(n-1)\cdots(l+1)}{(n-1-\ell)!} \left(\frac{1}{n}\right)^{n-1-\ell} \frac{n}{\ell} \\
& + 2e^2 \sum_{\ell=n/2+1}^{n-2} \frac{(n/2-1)\cdots(\ell-n/2+1)}{(n-1-\ell)!} \left(\frac{1}{n}\right)^{n-1-\ell} \frac{n}{(\ell-n/2)} \\
\leq & O(1) + e^2 \sum_{\ell=1}^{n/2} \frac{\ell+1}{(n-1-\ell)!\ell} + 2e^2 \sum_{\ell=n/2+1}^{n-2} \frac{(\ell-n/2+1)}{(n-1-\ell)!(\ell-n/2)} \\
\leq & O(1) + 2e^2 \sum_{\ell=1}^{n/2} \frac{1}{(n-1-\ell)!} + 4e^2 \sum_{\ell=n/2+1}^{n-2} \frac{1}{(n-1-\ell)!} = O(1).
\end{aligned}
$$

The lower bound d is $O(1)$, much looser than the actual hitting time $\Theta(n \ln n)$.

The two case studies reveal that existing lower bounds (Sudholt, 2012; Doerr and Kötzing, 2022) are loose for fitness landscapes with shortcuts.

4 Metric bounds

4.1 Drift analysis with fitness levels

We propose a new method which combines fitness levels with drift analysis for estimating te hitting time of elitist EAs. Its workflow is outlined below. For the sake of illustration, we only present the lower bound.

First, the search space S is split into multiple fitness levels $(S_0, \cdots, S_K)$ according to the fitness value from high to low, where $S_0 = S_{\text{opt}}$.

Secondly, states at the same level are assigned to the same distance from the optimal set, that is, for any $X \in S_0$, $d(X) = 0$ and for any $k \geq 1$ and $X \in S_k$, $d(X) = d_k$. The distance d_k is constructed based on a fitness level partition.

Next we need to prove that for any k and $X_k \in S_k$, d_k is a lower bound on the mean hitting time $m(X_k)$. Since an elitist EA never moves from $X_k \in S_k$ to a fitness level lower than S_k, the drift

$$
\Delta d(X_k) = d_k - \sum_{\ell=0}^{k} d_\ell p(X_k, S_\ell) = d_k p(X_k, S_{[0,k-1]}) - \sum_{\ell=1}^{k-1} d_\ell p(X_, S_\ell). \tag{31}
$$

According to Lemma 1, if for any $k \geq 1$ and $X_k \in S_k$, the drift $\Delta d(X_k) \leq 1$, then the hitting time $m(X_k) \geq d(X_k)$.

Finally, the tightest lower bound problem is regarded as a constrained multi-objective optimization problem subject to the constraint that d_k is constructed based on a fitness level partition.

The above drift analysis with fitness levels treats the fitness level method as a special kind of drift analysis. It is completely differs from existing fitness level methods (Wegener, 2003; Sudholt, 2012; Doerr and Kötzing, 2022).

4.2 Metric bounds

Using transition probabilities between fitness levels, we construct a lower bound d_k recursively by (32). According to Lemma 1, the drift $\Delta d(X_k) \leq 1$ ensures that d_k is a lower bound. The best lower bound d_k^* is reached when Inequality (32) becomes an equality.

Theorem 1 (Type-$r_{k,\ell}$ lower bound)**.** *Given an elitist EA for maximizing $f(x)$, a fitness level partition $(S_0, \cdots, S_K)$, probabilities $p(X_k, S_{[0,k-1]})$ and $r(X_k, S_\ell)$ (where $1 \le \ell < k \le K$), consider the family of distances $(d_1, \cdots, d_k)$ such that for any $X_k \in S_k$, $d(X_k) = d_k$. Then for any $k > 0$ and $X_k \in S_k$, the drift $\Delta d(X_k) \le 1$ if and only if*

$$d_k \le \min_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell \right\}. \tag{32}$$

Proof. First we prove the sufficient condition. Suppose that (32) is true. Since the EA is elitist, for any $k \ge 1$ and $X_k \in S_k$, by (31), we have

$$\Delta d(X_k) = p(X_k, S_{[0,k-1]}) d_k - \sum_{\ell=1}^{k-1} p(X_k, S_\ell) d_\ell.$$

We replace d_k (but not d_ℓ) with (32) and get

$$\begin{aligned}
\Delta d(X_k) &\le p(X_k, S_{[0,k-1]}) \min_{Y_k \in S_k} \left\{ \frac{1}{p(Y_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(Y_k, S_\ell) d_\ell \right\} - \sum_{\ell=1}^{k-1} p(X_k, S_\ell) d_\ell \\
&\le p(X_k, S_{[0,k-1]}) \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,k-1]}} d_\ell \right\} - \sum_{\ell=1}^{k-1} p(X_k, S_\ell) d_\ell \\
&= 1 + \sum_{\ell=1}^{k-1} p(X_k, S_\ell) d_\ell - \sum_{\ell=1}^{k-1} p(X_k, S_\ell) d_\ell = 1.
\end{aligned}$$

We complete the proof of the sufficient condition.

Secondly, we prove the necessary condition. Suppose that for any $k \ge 1$ and $X_k \in S_k$, $\Delta d(X_k) \le 1$. Since the EA is elitist, by (31), we have

$$\Delta d(X_k) = d_k p(X_k, S_{[0,k-1]}) - \sum_{\ell=1}^{k-1} d_\ell p(X_k, S_\ell) \le 1.$$

Then we have

$$d_k \le \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell. \tag{33}$$

Since the above inequality is true for all $X_k \in S_k$, it holds for the minimum over all X_k. We complete the proof of the necessary condition. □

Similarly, we construct an upper bound d_k recursively by (34). According to Lemma 2, the drift $\Delta d(X_k) \ge 1$ ensures that d_k is an upper bound. The best upper bound d_k^* is reached when Inequality (34) becomes an equality.

Theorem 2 (Type-$r_{k,\ell}$ upper bound)**.** *Given an elitist EA for maximizing $f(x)$, a fitness level partition $(S_0, \cdots, S_K)$, probabilities $p(X_k, S_{[0,k-1]})$ and $r(X_k, S_\ell)$ (where $1 \le \ell < k \le K$), consider the family of distances $(d_1, \cdots, d_k)$ such that for any $X_k \in S_k$, $d(X_k) = d_k$. Then for any $k \ge 1$ and $X_k \in S_k$, the drift $\Delta d(X_k) \ge 1$ if and only if*

$$d_k \ge \max_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell. \right\} \tag{34}$$

A challenge is how to quickly calculate d_k. In this paper, we will not calculate d_k recursively via (32) or (34). Instead, we convert metric bounds to linear bounds. For example, the upper bound (34) becomes

$$d_k = \frac{1}{p_{\min}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r_{\max}(X_k, S_\ell) d_\ell, \qquad k = 1, \cdots, K.$$

Then by induction, we represent d_k in a linear form as follows:

$$d_k = \frac{1}{p_{\min}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\min}(X_\ell, S_{[0,\ell-1]})}.$$

Thus the problem of calculating a metric bound becomes the problem of calculating a linear bound or coefficients $c_{k,\ell}$. This is discussed in detail in the next section.

4.3 The tightest metric bounds

Consider the problem of the tightest lower bound first. Given a family of lower bounds based on a fitness level partition, we want to determine which bound is the tightest. A family of lower bounds can be represented by a family of distances in drift analysis. Given a fitness level partition $(S_0, \cdots, S_K)$, consider the family of distances $(d_0, \cdots, d_K)$ such that $d_0 = 0$ and for any $k \geq 1$, the drift $\Delta d(X_k) \leq 1$. The condition $\Delta d(X_k) \leq 1$ ensures that d_k is a lower bound on the mean hitting time $m(X_k)$.

The tightest lower bound problem is a constrained multi-objective optimization problem:

$$\max\{d_k : \Delta d(X_k) \leq 1\}, \quad k = 1, \cdots, K, \tag{35}$$

subject to the constraint that $d_0 = 0$ and for all $k \geq 1$ and $X_k \in S_k$, $d(X_k) = d_k$. According to Theorem 1, the best lower bound d_k^* by (32) is the tightest.

Theorem 3. *Given an elitist EA for maximizing $f(x)$, a fitness level partition $(S_0, \cdots, S_K)$, probabilities $p(X_k, S_{[0,k-1]})$ and $r(X_k, S_\ell)$ (where $1 \leq \ell < k \leq K$), consider the family of distances $(d_0, d_1, \cdots, d_k)$ such that $d_0 = 0$ and for all $k \geq 1$ and $X_k \in S_k$, $d(X_k) = d_k$ and the drift $\Delta d(X_k) \leq 1$. The tightest lower bound within this distance family is*

$$d_k^* = \min_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell^* \right\}. \tag{36}$$

Proof. For $k = 1, \cdots, K$, since $\Delta d(X_k) \leq 1$, according to Theorem 1, we have

$$d_k \leq \min_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell \right\}.$$

The above d_k reaches the maximum when the inequality becomes an equality. □

Similarly, the tightest upper bound problem is another constrained multi-objective optimization problem:

$$\min\{d_k : \Delta d(X_k) \geq 1\}, \quad k = 1, \cdots, K. \tag{37}$$

subject to the constrain that $d_0 = 0$ and for all $k \geq 1$ and $X_k \in S_k$, $d(X_k) = d_k$. According to Theorem 2, the best upper bound d_k^* by (34) is the tightest.

Theorem 4. *Given an elitist EA for maximizing $f(x)$, a fitness level partition $(S_0, \cdots, S_K)$, probabilities $p(X_k, S_{[0,k-1]})$ and $r(X_k, S_\ell)$ (where $1 \leq \ell < k \leq K$), consider the family of distances $(d_0, \cdots, d_k)$ such that $d_0 = 0$ and for $k \geq 1$ and all $X_k \in S_k$, $d(X_k) = d_k$ and the drift $\Delta d(X_k) \geq 1$. The tightest upper bound within the distance family is*

$$d_k^* = \max_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell^* \right\}. \tag{38}$$

5 Linear bounds

5.1 Linear bounds

Based on metric bounds, we constructs linear bounds (1) and (2). The theorem below provides coefficients in the lower bound (1).

Theorem 5 (Type-$c_{k,\ell}$ lower bound)**.** *Given an elitist EA for maximizing $f(x)$, a fitness level partition $(S_0, \cdots, S_K)$, probabilities $p_{\max}(X_\ell, S_{[0,\ell-1]})$ and $r(X_k, S_\ell)$ (where $1 \leq \ell < k \leq K$), construct d_k by*

$$d_k = \frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})}, \tag{39}$$

where coefficients $c_{k,\ell} \in [0, 1]$ satisfy

$$c_{k,\ell} \leq \min_{X_k \in S_k} \left\{ r(X_k, S_\ell) + \sum_{j=\ell+1}^{k-1} r(X_k, S_j) c_{j,\ell} \right\}. \tag{40}$$

Then for any $k > 0$ and $X_k \in S_k$, the mean hitting time $m(X_k) \geq d_k$.

Proof. According to Lemma 1, it is sufficient to prove that for any $k \geq 1$ and $X_k \in S_k$, the drift $\Delta d(X_k) \leq 1$. Since the EA is elitist, from (31), we know

$$\Delta d(X_k) = p(X_k, S_{[0,k-1]}) d_k - \sum_{\ell=1}^{k-1} p(X_k, S_\ell) d_\ell = p(X_k, S_{[0,k-1]}) \left(d_k - \sum_{\ell=1}^{k-1} r(X_k, S_\ell) d_\ell \right).$$

We replace d_k and d_ℓ with (39) and get

$$\begin{aligned} \Delta d(X_k) = {} & p(X_k, S_{[0,k-1]}) \left[\frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})} \right] \\ & - p(X_k, S_{[0,k-1]}) \left[\sum_{\ell=1}^{k-1} r(X_k, S_\ell) \left(\frac{1}{p_{\max}(X_\ell, S_{[0,\ell-1]})} + \sum_{j=1}^{\ell-1} \frac{c_{\ell,j}}{p_{\max}(X_j, S_{[0,j-1]})} \right) \right]. \end{aligned} \tag{41}$$

In the double summation $\sum_{\ell=1}^{k-1} \sum_{j=1}^{\ell-1}$, the first term for $\ell = 1$ is empty but it is kept for the sake of notation. We expand this double summation and then merge the same term $1/p_{\max}(X_\ell, S_{[0,\ell-1]})$ (where $\ell = 1, \cdots, k$) as follows.

$$
\begin{aligned}
&\sum_{\ell=1}^{k-1} r(X_k, S_\ell) \sum_{j=1}^{\ell-1} \frac{c_{\ell,j}}{p_{\max}(X_j, S_{[0,j-1]})} = \sum_{\ell=2}^{k-1}\sum_{j=1}^{\ell-1} \frac{r(X_k, S_\ell) c_{\ell,j}}{p_{\max}(X_j, S_{[0,j-1]})} \\
=&\frac{r(X_k, S_2)c_{2,1}}{p_{\max}(X_1, S_0)} + \cdots + \left(\frac{r(X_k, S_{k-1})c_{k-1,1}}{p_{\max}(X_1, S_0)} + \frac{r(X_k, S_{k-1})c_{k-1,k-2}}{p_{\max}(X_{k-2}, S_{[0,k-3]})}\right) \\
=&\frac{\sum_{j=2}^{k-1} r(X_k, S_j)c_{j,1}}{p_{\max}(X_1, S_0)} + \cdots + \frac{\sum_{j=k-1}^{k-1} r(X_k, S_j)c_{j,k-2}}{p_{\max}(X_{k-2}, S_{[0,k-3]})} \\
=&\sum_{\ell=1}^{k-2} \frac{\sum_{j=\ell+1}^{k-1} r(X_k, S_j)c_{j,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})} = \sum_{\ell=1}^{k-1} \frac{\sum_{j=\ell+1}^{k-1} r(X_k, S_j)c_{j,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})}.
\end{aligned}
\tag{42}
$$

In the double summation $\sum_{\ell=1}^{k-1}\sum_{j=\ell+1}^{k-1}$, the last term for $\ell = k-1$ is empty but it is added for the sake of notation.

Inserting (42) into (41), we have

$$
\begin{aligned}
\Delta d(X_k) \leq & \, p(X_k, S_{[0,k-1]}) \left[\frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})}\right] \\
& - p(X_k, S_{[0,k-1]}) \left[\sum_{\ell=1}^{k-1}\left(\frac{r(X_k, S_\ell)}{p_{\max}(X_\ell, S_{[0,\ell-1]})} + \frac{\sum_{j=\ell+1}^{k-1} r(X_k, S_j)c_{j,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})}\right)\right].
\end{aligned}
\tag{43}
$$

According to Condition (40), $c_{k,\ell} \leq r(X_k, S_\ell) + \sum_{j=\ell+1}^{k-1} r(X_k, S_j)c_{j,\ell}$. Inserting it to (43), we get $\Delta d(X_k) \leq 1$ and complete the proof. □

Similarly, the theorem below provides coefficients in the upper bound (2). Its proof is similar to Theorem 5.

Theorem 6 (Type-$c_{k,\ell}$ upper bound)**.** *Given an elitist EA for maximizing $f(x)$, a fitness level partition $(S_0, \cdots, S_K)$, probabilities $p_{\min}(X_\ell, S_{[0,\ell-1]})$ and $r(X_k, S_\ell)$ where $1 \leq \ell < k \leq K$, construct d_k by*

$$
d_k = \frac{1}{p_{\min}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\min}(X_\ell, S_{[0,\ell-1]})}. \tag{44}
$$

where coefficients $c_{k,\ell} \in [0, 1]$ satisfy

$$
c_{k,\ell} \geq \max_{X_k \in S_k} \left\{ r(X_k, S_\ell) + \sum_{j=\ell+1}^{k-1} r(X_k, S_j)c_{j,\ell} \right\}. \tag{45}
$$

Then for any $k > 0$ and $X_k \in S_k$, the mean hitting time $m(X_k) \geq d_k$.

The best upper bound d_k^* and its coefficients $c_{k,\ell}^*$ are reached when Inequality (45) becomes an equality. The best lower bound d_k^* and its coefficients $c_{k,\ell}^*$ are reached when Inequality (40) becomes an equality.

There are three different ways to calculate coefficients $c_{k,\ell}$ by solving Inequality (40) or (45).

1. find an explicit expression of $c_{k,\ell}$ by (40) or (45);
2. recursively calculate $c_{\ell+1,\ell}, c_{\ell+2,\ell}, \cdots, c_{k,\ell}$ by (40) or (45);
3. combine the above two ways together, that is, for some $c_{k,\ell}$, use recursive calculations; but for other $c_{k,\ell}$, use an explicit expression.

5.2 Explicit expressions for linear bound coefficients

An explicit expression of coefficients $c_{k,\ell}$ is more convenient in applications. From (40), by induction, it is straightforward to obtain an explicit expression for lower bound coefficients as follows.

$$\begin{aligned} c_{k,\ell} \leq & r_{\min}(X_k, S_\ell) + \sum_{\ell<j_1<k} r_{\min}(X_k, S_{j_1})\, r_{\min}(X_{j_1}, S_\ell) + \cdots \\ & + \sum_{\ell<j_{k-\ell-1}<\cdots<j_1<k} r_{\min}(X_k, S_{j_1})\, r_{\min}(X_{j_1}, S_{j_2}) \cdots r_{\min}(X_{j_{k-\ell-1}}, S_\ell). \end{aligned} \tag{46}$$

Inequality (46) provides an intuitive interpretation of coefficient $c_{k,\ell}$. Each product in (46) can be interpreted as a conditional probability of the EA to visit S_ℓ starting from X_k. The coefficient $c_{k,\ell}$ is a lower bound on the conditional probabilities to visit S_ℓ starting from X_k.

Similarly, from (45), by induction, it is straightforward to obtain an explicit expression for upper bound coefficients as follows.

$$\begin{aligned} c_{k,\ell} \geq & r_{\max}(X_k, S_\ell) + \sum_{\ell<j_1<k} r_{\max}(X_k, S_{j_1})\, r_{\max}(X_{j_1}, S_\ell) + \cdots \\ & + \sum_{\ell<j_{k-\ell-1}<\cdots<j_1<k} r_{\max}(X_k, S_{j_1})\, r_{\max}(X_{j_1}, S_{j_2}) \cdots r_{\max}(X_{j_{k-\ell-1}}, S_\ell). \end{aligned} \tag{47}$$

The number of summation terms in (46) and (47) is up to $(k-\ell-1)!$. Therefore, it is intractable to calculate coefficients by (46) and (47). But there are many ways to construct explicit expressions that can be calculated in polynomial time. For example, for the lower bound, a simple expression from (46) is $c_{k,\ell} \leq r_{\min}(X_k, S_\ell)$. Recently He et al. (2023) propose a simplified version of (46) as shown below, which can be used to obtain tight lower bounds on fitness landscapes with shortcuts.

$$c_{k,l} \leq \prod_{i\in[\ell+1,k]} r_{\min}(X_i, S_{[\ell,i-1]}). \tag{48}$$

For the upper bound, an explicit expression from (47) is $c_{k,\ell} \geq r_{\max}(X_k, S_{[\ell,k-1]})$.

Another simple way is to assign $c_{k,\ell} = 0, 1, c$, or c_ℓ. Although Type-$0, 1$, c and c_ℓ bounds have been studied in (Wegener, 2003; Sudholt, 2012; Doerr and Kötzing, 2022), our proof is completely different. Furthermore, our Type-c upper and Type-c_ℓ lower bounds require weaker conditions. Hence, they are not exactly the same as those in (Sudholt, 2012; Doerr and Kötzing, 2022).

Let $c_{k,\ell} = 0$, then the linear lower bound (39) becomes the same Type-0 lower bound as Proposition 1.

Corollary 1 (Type-0 lower bound)**.** *For* $1 \leq \ell < k \leq K$, *choose* $c_{k,\ell} = 0$, *then* $m(X_k) \geq \frac{1}{p_{\max}(X_k, S_{[0,k-1]})}$.

Let $c_{k,\ell} = c$, then the linear lower bound (39) becomes a Type-c lower bound.

Corollary 2 (Type-c lower bound)**.** *For* $1 \leq \ell < k \leq K$, *choose* $c_{k,\ell} = c$ *to satisfy the inequality*

$$c \leq \min_{1<k\leq K} \min_{1\leq \ell<k} \min_{X_k: p(X_k, S_{[0,\ell]})>0} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})}. \tag{49}$$

Then $m(X_k) \geq \frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c}{p_{\max}(X_\ell, S_{[0,\ell-1]})}$.

Proof. Condition (49) is equivalent to that for any $1 \leq \ell < k \leq K$ and $X_k \in S_k$ such that $p(X_k, S_{[0,\ell-1]}) > 0$, it holds

$$c \leq \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})} = \frac{r(X_k, S_\ell)}{r(X_k, S_{[0,\ell]})}.$$

$$c\, r(X_k, S_{[0,\ell]}) = c\,(1 - r(X_k, S_{[\ell+1,k-1]})) \leq r(X_k, S_\ell). \tag{50}$$

$$c \leq r(X_k, S_\ell) + c\, r(X_k, S_{[\ell+1,k-1]}). \tag{51}$$

For any $1 \leq \ell < k \leq K$ and $X_k \in S_k$ such that $p(X_k, S_{[0,\ell-1]}) = 0$, we have $r(X_k, S_{[0,\ell]}) = 0$ and $r(X_k, S_{[\ell+1,k-1]}) = 1$. Thus we get an identical equation:

$$c = r(X_k, S_\ell) + c\, r(X_k, S_{[\ell+1,k-1]}) = 0 + c. \tag{52}$$

Combining (51) and (52), we get Condition (40), that is for all $X_k \in S_k$,

$$c \leq r(X_k, S_\ell) + c \sum_{j=\ell+1}^{k-1} r(X_k, S_\ell).$$

The above inequality is true for the minimum over all X_k. According to Theorem 5, we get the corollary. □

Corollary 2 is more convenient than Proposition 3 because the coefficient c is calculated directly by probabilities $p(X_k, S_\ell)$ and $p(X_k, S_{[0,\ell]})$. Inequality (50) is equivalent to the inequality $r_{k,\ell} \geq c\sum_{j=0}^{\ell} r_{k,j}$ in Proposition 3 under different representations, so Corollary 2 is equivalent to Proposition 3.

Let $c_{k,\ell} = c_\ell$, then the linear lower bound (39) becomes a Type-c_ℓ lower bound. The proof of Corollary 3 is similar to Corollary 2 so we omit its proof.

Corollary 3 (Type-c_ℓ lower bound)**.** *For $1 \leq \ell < k \leq K$, choose $c_{k,\ell} = c_\ell$ to satisfy the inequality*

$$c_\ell \leq \min_{\ell < k \leq K} \min_{X_k : p(X_k, S_{[0,\ell]}) > 0} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})}. \tag{53}$$

Then $m(X_k) \geq \frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_\ell}{p_{\max}(X_\ell, S_{[0,\ell-1]})}$.

The above corollary is equivalent to Lemma 3 without the initialization condition (12). Corollary 3 can be used to handle random initialization by replacing X_k with $X^{[0]}$ and the mean hitting time

$$m \geq \sum_{k=1}^{K} \Pr(X^{[0]} \in S_k) \left(\frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_\ell}{p_{\max}(X_\ell, S_{[0,\ell-1]})} \right).$$

Similarly, let $c_{k,\ell} = 1$, then linear upper bound (44) becomes the same Type-1 bound as in Proposition 2.

Corollary 4 (Type-1 upper bound)**.** *For $1 \leq \ell < k \leq K$, choose $c_{k,\ell} = 1$, then* $m(X_k) \leq \sum_{\ell=1}^{k} \frac{1}{p_{\min}(X_\ell, S_{[0,\ell-1]})}$.

Let $c_{k,\ell} = c$, then the linear lower bound (44) becomes a Type-c upper bound. The proof of Corollary 5 is similar to Corollary 2. We omit its proof since we only need to replace the minimum with the maximum.

Corollary 5 (Type-c upper bound). *For $1 \leq \ell < k \leq K$, choose $c_{k,\ell} = c$ to satisfy*

$$c \geq \max_{1<k\leq K} \max_{1\leq \ell<k} \max_{X_k: p(X_k, S_{[0,\ell]})>0} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})}. \tag{54}$$

Then $m(X_k) \leq \frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c}{p_{\max}(X_\ell, S_{[0,\ell-1]})}$.

Corollary 5 is more convenient than Proposition 4 because the coefficient c is calculated directly by transition probabilities $p(X_k, S_\ell)$ and $p(X_k, S_{[0,\ell]})$. Unlike Proposition 4, Corollary 5 does not require the condition that for all $1 \leq l \leq K-2$, $(1-c)p_{\min}(X_{\ell+1}, S_{[0,\ell]}) \leq p_{\min}(X_\ell, S_{[0,\ell-1]})$. Therefore, Proposition 4 is a special case of Corollary 5.

Let $c_{k,\ell} = c_\ell$, then the linear upper bound (44) becomes a Type-c_ℓ upper bound. The proof of Corollary 6 is similar to Corollary 2.

Corollary 6 (Type-c_ℓ upper bound). *For $1 \leq \ell < k \leq K$, choose $c_{k,\ell} = c_\ell$ to satisfy*

$$c_\ell \geq \max_{\ell<k<K} \max_{X_k: p(X_k, S_{[0,\ell]})>0} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})}. \tag{55}$$

then $m(X_k) \leq \frac{1}{p_{\min}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_\ell}{p_{\min}(X_\ell, S_{[0,\ell-1]})}$.

Corollary 6 is new and completely different from Proposition 6. Coefficients c_ℓ are directly calculated by transition probabilities $p(X_k, S_\ell)$ and $p(X_k, S_{[0,\ell]})$.

5.3 Discussion of different linear bounds

As shown in Corollaries 1 to 6, Types-$0, 1, c, c_\ell$ bounds are special cases of Type-$c_{k,\ell}$ bounds. Therefore, Type-$c_{k,\ell}$ bounds are the best among them.

Corollary 7. *Given an elitist EA and a fitness level partition $(S_0, \cdots, S_K)$, the best Type-$c_{k,\ell}$ bound is no worse than the best Type-c and Type-c_ℓ bounds for lower and upper bounds.*

The best Type-$c_{k,\ell}$ bound is the exact hitting time of EAs on level-based fitness landscapes, but the best Type-c and c_ℓ bounds usually are not.

Definition 5. *Given an elitist EA for maximizing a function $f(x)$ and a fitness level partition $(S_0, \cdots, S_K)$, we call $f(x)$ a* level-based landscape *for the EA if for all $1 \leq \ell < k \leq K$ and $X_k \in S_k$, $p_{\min}(X_k, S_\ell) = p_{\max}(X_k, S_\ell)$.*

Both OneMax and TwoMax1 are level-based fitness landscapes for the (1+1) EA. Corollary 8 follows directly from Theorem 5 and Theorem 6.

Corollary 8. *Given an elitist EA for maximizing a function $f(x)$ and a fitness level partition $(S_0, \cdots, S_K)$, if $f(x)$ is a a level-based fitness landscape for the EA, then the best Type-$c_{k,\ell}$ lower bound is equal to the best Type-$c_{k,\ell}$ upper bound.*

In addition, Type-c and Type-c_ℓ lower bounds are loose on fitness landscapes with shortcuts because shortcuts results in coefficients c and c_ℓ as small as $o(1)$. we have discussed this issue in Subsection 3.2 and Subsection 3.3. We obtain a more generally conclusion as follows.

Theorem 7. *If a shortcut exists, that is, for some $1 \leq \ell < k \leq K$ and $X_k \in S_k$, it holds*

$$\frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})} = o(1), \tag{56}$$

then coefficients $c = o(1)$ in (49) and $c_\ell = o(1)$ in (53).

Proof. According to Corollary 2, the lower bound coefficient

$$c \leq \min_{1<k\leq K} \min_{1\leq \ell<k} \min_{X_k: p(X_k, S_{[0,\ell]})>0} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})}.$$

By Condition (56), we get $c = o(1)$.

According to Corollary 3, the lower bound coefficient

$$c_\ell \leq \min_{\ell<k\leq K} \min_{X_k: p(X_k, S_{[0,\ell]})>0} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,\ell]})}.$$

By Condition (56), we get $c_\ell = o(1)$. □

6 Applications

6.1 Case Study 3: Calculating lower bound coefficients for the (1+1) EA on OneMax

In this case study, we demonstrate different ways to calculate coefficients in the linear lower bound. Consider the (1+1) EA on OneMax. Assume that the EA starts from S_n. According to Theorem 5 and Corollary 8, we get a Type-$c_{k,\ell}$ lower bound

$$d_n = \frac{1}{p(x_n, S_{[0,n-1]})} + \sum_{\ell=1}^{n-1} \frac{c_{n,\ell}}{p(x_\ell, S_{[0,\ell-1]})}.$$

where

$$c_{n,\ell} \leq r(x_n, S_\ell) + \sum_{k=\ell+1}^{n-1} r(x_n, S_k) c_{k,\ell}, \quad \ell = 1, \cdots, n-1. \tag{57}$$

Since OneMax is a level-based fitness landscape to the (1+1) EA, the best Type-$c_{k,\ell}$ bound is the exact hitting time. By (46), we obtain the best coefficient

$$\begin{aligned} c^*_{n,\ell} =& r(X_n, S_\ell) + \sum_{\ell<j_1<n} r(X_n, S_{j_1})\, r(X_{j_1}, S_\ell) + \cdots \\ &+ \sum_{\ell<j_{n-\ell-1}<\cdots<j_1<n} r(X_n, S_{j_1})\, r(X_{j_1}, S_{j_2}) \cdots r(X_{j_{n-\ell-1}}, S_\ell). \end{aligned} \tag{58}$$

Unfortunately the calculation of (58) is intractable.

Recall that $c_{n,\ell} \leq 1$. It is sufficient to compute large coefficients $c_{n,\ell} = \Omega(1)$ since a tight lower bound can be generated by large coefficients.

There are two approaches to calculate $c_{k,\ell}$ via Inequality (57). One is to look for explicit solutions to Inequality (57). The other is to recursively calculate $c_{k,\ell}$ level by level. There are different explicit solutions to Inequality (57). It is trivial to get the trivial solution $c_{n,\ell} = 0$ (where $1 \leq \ell \leq n-1$). From (58), it is straightforward to obtain an explicit solution

$$c_{n,\ell} = r(x_n, S_\ell) = \frac{p(x_n, S_\ell)}{p(x_n, S_{[0,n-1]})}.$$

Since $p(x_n, S_\ell) \geq \binom{n}{n-\ell} \left(\frac{1}{n}\right)^{n-\ell} \left(1-\frac{1}{n}\right)^\ell$ and $p(x_n, S_{[0,n-1]}) \leq 1$, we have

$$c_{n,\ell} \geq \binom{n}{n-\ell} \left(\frac{1}{n}\right)^{n-\ell} \left(1-\frac{1}{n}\right)^\ell = O\left(\frac{1}{(n-\ell)!}\right).$$

However, these coefficients are too small because $c_{n,\ell} = O\left(\frac{1}{(n-\ell)!}\right)$.

A non-trivial explicit solution is to let $c_{k,\ell} = c$. According to Corollary 2, we choose

$$c = \min_{1<k\leq n} \min_{1\leq \ell<k} \frac{p(x_k, S_\ell)}{p(x_k, S_{[0,\ell]})}. \tag{59}$$

Transition probabilities $p(x_k, S_\ell)$ (where $\ell = 1, \cdots, k-1$) are calculated as follows. Since x_k has k zero-valued bits, the transition from x_k to S_ℓ happens if $k-\ell$ zero-valued bits are flipped and other bits unchanged. Thus

$$p(x_k, S_\ell) \geq \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} \left(1-\frac{1}{n}\right)^{n-k+\ell}. \tag{60}$$

The transition from x_k to $S_{[0,\ell]}$ happens only if $k-\ell$ zero-valued bits are flipped. Thus

$$p(x_k, S_{[0,\ell]}) \leq \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell}. \tag{61}$$

Inserting (60) and (61) into (59), we have

$$c \geq \min_{1<k\leq n} \min_{1\leq \ell<k} \left(1-\frac{1}{n}\right)^{n-k+\ell} \geq e^{-1}.$$

Therefore, coefficients $c = \Omega(1)$.

Another non-trivial explicit solution is to let $c_{k,\ell} = c_\ell$. According to Corollary 3, we choose

$$c_\ell = \min_{\ell<k\leq n} \frac{p(x_k, S_\ell)}{p(x_k, S_{[0,\ell]})}. \tag{62}$$

Inserting (60) and (61) into (62), we have

$$c_\ell \geq \min_{\ell<k\leq n} \left(1-\frac{1}{n}\right)^{n-k+\ell} \geq e^{-1}.$$

Therefore, coefficients $c_\ell = \Omega(1)$.

Inequality (57) can be solved recursively from $k = \ell+1$ to n. A recursive solution to (57) is calculated as follows. According to (40), we choose

$$c_{\ell+1,\ell} = r(x_{\ell+1}, S_\ell) = \frac{p(x_{\ell+1}, S_\ell)}{p(x_{\ell+1}, S_{[0,\ell]})} \geq \left(1-\frac{1}{n}\right)^{n-1} \geq e^{-1} \quad \text{(by (60) and (61))}.$$

Assume that $c_{\ell+1,\ell}, \cdots, c_{k-1,\ell} \geq e^{-1}$. According to (40), we choose

$$\begin{aligned} c_{k,\ell} &= r(x_k, S_\ell) + \sum_{j=\ell+1}^{k-1} r(x_k, S_j) c_{j,\ell} \geq r(x_k, S_\ell) + r(x_k, S_{[\ell+1,k-1]}) e^{-1} \\ &= r(x_k, S_\ell) + [1 - r(x_k, S_{[0,\ell]})] e^{-1} = \frac{p(x_k, S_\ell) - p(x_k, S_{[0,\ell]})}{p(x_k, S_{[0,k-1]})} e^{-1} + e^{-1}. \end{aligned}$$

By (60) and (61), we have

$$p(x_k, S_\ell) - p(x_k, S_{[0,\ell]}) \geq \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} \left(1-\frac{1}{n}\right)^{n-k+\ell} - \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} e^{-1} \geq 0.$$

We get $c_{k,\ell} \geq e^{-1}$. By induction, $c_{k,\ell} \geq e^{-1}$ for $k = \ell+1, \cdots, n$. Thus, coefficients $c_{n,\ell} = \Omega(1)$.

Finally, we use a mixture of recursive and explicit solutions, that is, some coefficients are calculated recursively and some coefficients come from an explicit solution. For example, a mix of explicit and recursive solutions is

$$c_{k,\ell} = c \leq r(x_k, S_\ell) + \sum_{j=\ell+1}^{k-1} r(x_k, S_j)c, \qquad 1 \leq \ell < k \leq n-1. \tag{63}$$

$$c_{n,\ell} = r(x_n, S_\ell) + \sum_{j=\ell+1}^{n-1} r(x_n, S_j)c, \qquad 1 \leq \ell \leq n-1. \tag{64}$$

Similar to the analysis of the explicit solution c, for (63), we get $c = \Omega(1)$. In (64),

$$r(x_n, S_j) = \frac{p(x_n, S_j)}{p(x_n, S_{[0,n-1]})} \geq \frac{p(x_n, S_j)}{1} \geq \binom{n}{n-j} \left(\frac{1}{n}\right)^{n-j} \left(1-\frac{1}{n}\right)^j \quad \text{(by (60))},$$

then we get for $\ell = 1, \cdots, n-1$

$$c_{n,\ell} \geq \binom{n}{n-\ell} \left(\frac{1}{n}\right)^{n-\ell} \left(1-\frac{1}{n}\right)^\ell + e^{-1} \sum_{j=\ell+1}^{n-1} \binom{n}{n-j} \left(\frac{1}{n}\right)^{n-j} \left(1-\frac{1}{n}\right)^j.$$

Thus we get $c_{n,\ell} = \Omega(1)$.

In summary, there exist different ways to calculate coefficients $c_{k,\ell}$ from a trivial coefficient 0 to the exact coefficients $c^*_{k,\ell}$. Drift analysis with fitness levels is a framework that can be used to develop different fitness level methods. In particular, existing fitness level methods (Wegener, 2003; Sudholt, 2012; Doerr and Kötzing, 2022; He et al., 2023) can be regarded as special cases within the framework.

6.2 Case Study 4: A tight Type-$c_{k,\ell}$ lower bound of the (1+1) on TwoMax1

In this case study, we show that the Type-$c_{k,\ell}$ lower bound by Theorem 5 is tight on fitness landscapes with shortcuts. Consider the (1+1) EA maximizing TwoMax1. Assume that the EA starts from S_{n-1}. We prove that the Type-$c_{k,\ell}$ lower bound by Theorem 5,

$$d_{n-1} = \sum_{\ell=1}^{n-2} \frac{c_{n-1,\ell}}{p(x_\ell, S_{[0,\ell-1]})} \geq \sum_{\ell=1}^{n/2} \frac{c_{n-1,\ell}}{p(x_\ell, S_{[0,\ell-1]})}, \tag{65}$$

is $\Omega(n \ln n)$.

Transition probabilities $p(x_\ell, S_{[0,\ell-1]})$ (where $1 \leq \ell \leq n/2$) are calculated as follows. Since x_ℓ has ℓ zero-valued bits and $n-\ell$ one-valued bits, the transition from x_ℓ to $S_{[0,\ell-1]}$ happens only if either (i) 1 zero-valued bit is flipped, or (ii) x_ℓ is mutated to $(0, \cdots, 0)$. The probability of the first event is $\binom{\ell}{1}\frac{1}{n}$. The probability of the second event happening is $\left(\frac{1}{n}\right)^{n-\ell}\left(1-\frac{1}{n}\right)^\ell$. Thus

$$p(x_\ell, S_{[0,\ell-1]}) \leq \frac{\ell}{n} + \left(\frac{1}{n}\right)^{n-\ell} \left(1-\frac{1}{n}\right)^\ell \leq \frac{\ell+1}{n}. \tag{66}$$

Then we get

$$d_{n-1} \geq \sum_{\ell=1}^{n/2} c_{n-1,\ell} \frac{n}{\ell+1}. \tag{67}$$

According to Theorem 5, we choose coefficients

$$c_{n-1,\ell} = \sum_{k=\ell+1}^{n/2} r(x_{n-1}, S_k) c_{k,\ell}, \quad \ell = 1, \cdots, \frac{n}{2}. \tag{68}$$

Coefficients $c_{k,\ell}$ (where $1 \leq \ell < k \leq n/2$) are calculated using a constant c. According to Theorem 5, for $1 \leq \ell < k \leq n/2$, we choose coefficients $c_{k,\ell} = c$ such that

$$\begin{aligned} c &\leq r(x_k, S_\ell) + \sum_{i=\ell+1}^{k-1} r(x_k, S_i) c, \\ c &\leq \frac{r(x_k, S_\ell)}{1 - r(x_k, S_{[\ell+1,k-1]})} = \frac{r(x_k, S_\ell)}{r(x_k, S_{[0,\ell]})} = \frac{p(x_k, S_\ell)}{p(x_k, S_{[0,\ell]})}. \end{aligned}$$

The above inequality is true for all for $1 \leq \ell < k \leq n/2$, thus we choose

$$c = \min_{1 < k \leq n/2} \min_{1 \leq \ell < k} \frac{p(x_k, S_\ell)}{p(x_k, S_{[0,\ell]})}. \tag{69}$$

The above c is calculated as follows. Since x_k (where $1 \leq \ell < k \leq n/2$) has k zero-valued bits, the transition from x_k to S_ℓ happens if $k - \ell$ zero-valued bits are flipped and other bits are not flipped. Thus

$$p(x_k, S_\ell) \geq \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} \left(1 - \frac{1}{n}\right)^{n-k+\ell} \geq \binom{k}{k-l} \left(\frac{1}{n}\right)^{k-\ell} e^{-1}. \tag{70}$$

The transition from x_k to $S_{[0,\ell]}$ happens only if either (i) $k - \ell$ zero-valued bits are flipped, or (ii) x_k is mutated to $(0, \cdots, 0)$. The probability of the first event is $\binom{k}{k-\ell}(\frac{1}{n})^{k-\ell}$. The probability of the second event is $\left(\frac{1}{n}\right)^{n-k} \left(1 - \frac{1}{n}\right)^k$. Because $1 \leq \ell < k \leq n/2$, we have $n - k \geq n/2 \geq k - \ell + 1$. Thus

$$p(x_k, S_{[0,\ell]}) \leq \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} + \left(\frac{1}{n}\right)^{n-k} \leq \binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} + \left(\frac{1}{n}\right)^{k-\ell+1}. \tag{71}$$

Inserting (70) and (71) to (69), we get

$$c \geq \frac{\binom{k}{k-l} \left(\frac{1}{n}\right)^{k-\ell} e^{-1}}{\binom{k}{k-\ell} \left(\frac{1}{n}\right)^{k-\ell} + \left(\frac{1}{n}\right)^{k-\ell+1}} = \Omega(1).$$

Next coefficients $c_{n-1,\ell}$ (where $1 \leq \ell \leq n/2$) are calculated by (68). Inserting $c = \Omega(1)$ into (68), we get for $\ell = 1, \cdots, n/2$,

$$c_{n-1,\ell} \geq \Omega(1) r(x_{n-1}, S_{[\ell+1,n/2]}). \tag{72}$$

The conditional probability $r(x_{n-1}, S_{[\ell+1,n/2]})$ is calculated as follows. Since x_{n-1} has $n/2 + 1$ zero-valued bits, the transition from x_{n-1} to $S_{n/2}$ happens if 1 zero-valued bit is flipped and other bits are unchanged. Thus

$$p(x_{n-1}, S_{[\ell+1,n/2]}) \geq \binom{n/2+1}{1} \frac{1}{n} \left(1 - \frac{1}{n}\right)^{n-1} \geq \frac{1}{2e}.$$

Then we get

$$r(x_{n-1}, S_{[\ell,n/2]}) = \frac{p(x_{n-1}, S_{[\ell,n/2]})}{p(x_{n-1}, S_{[0,n-2]})} \geq p(x_{n-1}, S_{[\ell,n/2]}) \geq \frac{1}{2e}. \tag{73}$$

Inserting (73) into (72), we get $c_{n-1,\ell} = \Omega(1)$. Inserting $c_{n-1,\ell} = \Omega(1)$ into (67), we get

$$d_{n-1} \geq \Omega(1) \sum_{\ell=1}^{n/2-1} \frac{n}{\ell+1} = \Omega(n \ln n). \tag{74}$$

The lower bound d_{n-1} is tight since the actual hitting time is $\Theta(n \ln n)$. Table 2 compares Type-c, c_ℓ and $c_{k,\ell}$. lower bounds. The Type $c_{k,\ell}$ lower bound is tight but Type-c and c_ℓ lower bounds are not.

Table 2: Comparison of different types of lower bounds of the (1+1) EA on TwoMax1

Type-$c_{k,\ell}$	Type-c	Type-c_l
$\Omega(n \ln n)$	$O(1)$	$O(1)$
by Theorem 5	by Proposition 3	by Proposition 5 and Lemma 3

7 Conclusions

In this paper, we combine drift analysis with fitness levels to construct metric bounds using transition probabilities between fitness levels and generate the tightest metric bounds. From metric bounds, we derive general linear bounds and propose different methods to calculate coefficients in linear lower bounds. Drift analysis with fitness levels is a framework that can be used to develop different fitness level methods for different types of bounds. Table 3 summarizes the main bounds discussed in the paper.

Table 3: Type-$c_{k,\ell}$, c_ℓ and c bounds. Notation refers to Table 1.

Type	d_k: a bound on the hitting time $m(X_k)$	Source
$r_{k,\ell}$ lower	$d_k \leq \min_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,k-1]})} d_\ell \right\}$	Theorems 1,3
$r_{k,\ell}$ upper	$d_k \geq \max_{X_k \in S_k} \left\{ \frac{1}{p(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{p(X_k, S_\ell)}{p(X_k, S_{[0,k-1]})} d_\ell \right\}$	Theorems 2, 4
$c_{k,\ell}$ lower	$d_k \leq \frac{1}{p_{\max}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\max}(X_\ell, S_{[0,\ell-1]})}$	Theorem 5
$c_{k,\ell}$ upper	$d_k \geq \frac{1}{p_{\min}(X_k, S_{[0,k-1]})} + \sum_{\ell=1}^{k-1} \frac{c_{k,\ell}}{p_{\min}(X_\ell, S_{[0,\ell-1]})}$	Theorem 6
c_ℓ	$c_{k,\ell} = c_\ell$	Corollaries 3, 6
c	$c_{k,\ell} = c$	Corollaries 2, 5

The framework is generic and promising. It turns out that Type-$c_{k,\ell}$ bounds are at least as tight as Type-c_ℓ and Type-c bounds on any fitness landscapes, and even tighter on fitness landscapes with shortcuts. This is demonstrated by the case study of the (1+1) EA maximizing the TwoMax1 function. One direction for future research is to simplify the recursive computation in metric and linear bounds.

Acknowledgments

The authors thank Dirk Sudholt for his helpful discussion of the fitness level method, and also Benjamin Doerr and Timo Kötzing for their kind explanations of their work, which helped to greatly improve the quality of this paper.

References

Aboutaib, B. and Sutton, A. M. (2022). Runtime analysis of unbalanced block-parallel evolutionary algorithms. In *International Conference on Parallel Problem Solving from Nature*, pages 555–568. Springer.

Antipov, D., Buzdalova, A., and Stankevich, A. (2018). Runtime analysis of a population-based evolutionary algorithm with auxiliary objectives selected by reinforcement learning. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 1886–1889.

Case, B. and Lehre, P. K. (2020). Self-adaptation in nonelitist evolutionary algorithms on discrete problems with unknown structure. *IEEE Transactions on Evolutionary Computation*, 24(4):650–663.

Chen, T., He, J., Sun, G., Chen, G., and Yao, X. (2009). 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.

Corus, D., Dang, D.-C., Eremeev, A. V., and Lehre, P. K. (2017). Level-based analysis of genetic algorithms and other search processes. *IEEE Transactions on Evolutionary Computation*, 22(5):707–719.

Corus, D., Oliveto, P. S., and Yazdani, D. (2020). When hypermutations and ageing enable artificial immune systems to outperform evolutionary algorithms. *Theoretical Computer Science*, 832:166–185.

Doerr, B. (2019). Analyzing randomized search heuristics via stochastic domination. *Theoretical Computer Science*, 773:115–137.

Doerr, B. and Goldberg, L. A. (2013). Adaptive drift analysis. *Algorithmica*, 65(1):224–250.

Doerr, B., Johannsen, D., and Winzen, C. (2012). Multiplicative drift analysis. *Algorithmica*, 4(64):673–697.

Doerr, B. and Kötzing, T. (2022). Lower bounds from fitness levels made easy. *Algorithmica*.

Doerr, B. and Neumann, F. (2019). *Theory of evolutionary computation: Recent developments in discrete optimization*. Springer Nature.

He, J., Chen, T., and Yao, X. (2015). On the easiest and hardest fitness functions. *IEEE Transactions on Evolutionary Computation*, 19(2):295–305.

He, J., Chong, S. Y., and Yao, X. (2023). Fast estimations of lower bounds on hitting time of elitist evolutionary algorithms. *arXiv*. doi.org/10.48550/arXiv.2311.10502.

He, J. and Lin, G. (2016). Average convergence rate of evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 20(2):316–321.

He, J. and Yao, X. (2001). Drift analysis and average time complexity of evolutionary algorithms. *Artificial Intelligence*, 127(1):57–85.

He, J. and Yao, X. (2002). 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.

He, J. and Yao, X. (2003). Towards an analytic framework for analysing the computation time of evolutionary algorithms. *Artificial Intelligence*, 145(1-2):59–97.

He, J. and Yao, X. (2017). Average drift analysis and population scalability. *IEEE Transactions on Evolutionary Computation*, 21(3):426–439.

Huang, H., Su, J., Zhang, Y., and Hao, Z. (2019). An experimental method to estimate running time of evolutionary algorithms for continuous optimization. *IEEE Transactions on Evolutionary Computation*, 24(2):275–289.

Knuth, D. E. (1976). Big omicron and big omega and big theta. *ACM Sigact News*, 8(2):18–24.

Kötzing, T. and Krejca, M. S. (2019). First-hitting times under drift. *Theoretical Computer Science*, 796:51–69.

Lengler, J. (2020). Drift analysis. In *Theory of Evolutionary Computation: Recent Developments in Discrete Optimization*, pages 89–131. Springer.

Malalanirainy, T. and Moraglio, A. (2022). Runtime analysis of convex evolutionary search algorithm with standard crossover. *Swarm and Evolutionary Computation*, 71:101078.

Oliveto, P. S., He, J., and Yao, X. (2007). Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results. *International Journal of Automation and Computing*, 4(3):281–293.

Oliveto, P. S., Sudholt, D., and Witt, C. (2022). Tight bounds on the expected runtime of a standard steady state genetic algorithm. *Algorithmica*, 84(6):1603–1658.

Oliveto, P. S. and Witt, C. (2011). Simplified drift analysis for proving lower bounds in evolutionary computation. *Algorithmica*, 59(3):369–386.

Quinzan, F., Göbel, A., Wagner, M., and Friedrich, T. (2021). Evolutionary algorithms and submodular functions: benefits of heavy-tailed mutations. *Natural Computing*, 20(3):561–575.

Rajabi, A. and Witt, C. (2020). Self-adjusting evolutionary algorithms for multimodal optimization. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, pages 1314–1322.

Sudholt, D. (2012). A new method for lower bounds on the running time of evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 17(3):418–435.

Wegener, I. (2001). Theoretical aspects of evolutionary algorithms. In *International Colloquium on Automata, Languages, and Programming*, pages 64–78. Springer.

Wegener, I. (2003). Methods for the analysis of evolutionary algorithms on pseudo-Boolean functions. In *Evolutionary optimization*, pages 349–369. Springer.

Witt, C. (2014). Fitness levels with tail bounds for the analysis of randomized search heuristics. *Information Processing Letters*, 114(1-2):38–41.

Yu, Y. and Zhou, Z.-H. (2008). A new approach to estimating the expected first hitting time of evolutionary algorithms. *Artificial Intelligence*, 172(15):1809–1832.

Zhou, Y., He, J., and Nie, Q. (2009). A comparative runtime analysis of heuristic algorithms for satisfiability problems. *Artificial Intelligence*, 173(2):240–257.