More Precise Runtime Analyses of Non-elitist EAs in Uncertain Environments

Per Kristian Lehre*
University of Birmingham
Birmingham, United Kingdom
p.k.lehre@cs.bham.ac.uk

Xiaoyu Qin*
University of Birmingham
Birmingham, United Kingdom
xxq896@cs.bham.ac.uk

ABSTRACT

Real-world optimisation problems often involve uncertainties. In the past decade, several rigorous analysis results for evolutionary algorithms (EAs) on discrete problems show that EAs can cope with low-level uncertainties, and sometimes benefit from uncertainties. Using non-elitist EAs with large population size is a promising approach to handle higher levels of uncertainties. However, the performance of non-elitist EAs in some common fitness-uncertainty scenarios is still unknown.

We analyse the runtime of non-elitist EAs on OneMax and LeadingOnes under prior and posterior noise models, and the dynamic binary value problem (DynBV). Our analyses are more extensive and precise than previous analyses of non-elitist EAs. In several settings, we prove that the non-elitist EAs beat the current state of the art results. Previous work shows that the population size and mutation rate can dramatically impact the performance of non-elitist EAs. The optimal choices of these parameters depend on the level of uncertainties in the fitness functions. We provide more precise guidance on how to choose mutation rate and population size as a function of the level of uncertainties.

CCS CONCEPTS

• **Theory of computation → Optimisation with randomised search heuristics**.

KEYWORDS

Non-elitist evolutionary algorithms, uncertain environments, noisy optimisation, dynamic optimisation, runtime analysis

ACM Reference Format:
Per Kristian Lehre and Xiaoyu Qin. 2021. More Precise Runtime Analyses of Non-elitist EAs in Uncertain Environments. In *2021 Genetic and Evolutionary Computation Conference (GECCO '21), July 10–14, 2021, Lille, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3449639.3459312

1 INTRODUCTION

In real-world optimisation problems, a wide range of uncertainties has to be considered [25]. Runtime analyses of EAs have considered four abstracted uncertainty models which are prior noise, posterior noise, dynamic environment and partial evaluation. The prior noise randomly flips one or several bits before each evaluation, e.g. one-bit noise [9], which flips one bit uniformly with probability q, and bit-wise noise [21], which flips each bit independently with probability p, while the posterior noise, also called additive noise, adds a random value after each evaluation, e.g. Gaussian noise [13], which adding a value sampled from $\mathcal{N}\left(0, \sigma^2\right)$. In dynamic optimisation, the fitness function is fixed in each generation but is varied by time, e.g. the noisy linear function [18] and DynBV problem [16].

The simple (1+1) EA is robust to some uncertainty but also can be inefficient under high-level uncertainties. For example, the runtime of the (1+1) EA on LeadingOnes is exponential if $q = \Omega(1)$ and $p = \Omega\left(1/n^2\right)$ under one-bit noise and bit-wise noise respectively [13, 21, 23]. Some algorithms can furthermore improve the robustness, such as estimation of distribution algorithms (EDAs) [10, 15], ant colony optimisation (ACO) [12], the (1+1) EA using resampling strategy [21, 22] and population-based algorithms [3, 4, 13, 20]. For the efficiency of non-elitist EAs under uncertainties, Dang and Lehre [3] showed that binary tournament selection with a sufficient population size and a conservative mutation rate has expected runtime $O(n\log(n)\log(\log(n)))$ on OneMax under any one-bit noise level. They also proved that the non-elitist EA can handle extremely high-level additive noises [3] and partial evaluation optimisation [4]. However, the robustness of non-elitist EAs to noise is still unknown in several settings. Tables 1-6 summarise recent theoretical studies (including this paper) of EAs in noisy settings. Note that some previous studies do not contain exact runtimes. In these cases, the runtime results are deduced from the proofs. For the robustness in the dynamic environments, Lengler and Schaller [18] proved that the (1+1) EA can optimise the random weights linear function, in which the weights in each generation are randomly sampled, in $O(n\log(n))$ times. However, the efficiency of population-based algorithms in this setting is currently unknown, though Lengler and Riedi [17] analysed the $(\mu + 1)$ EA on the DynBV assuming that the population is initialised close to the optimum.

Parameter configuration can dramatically impact the performance of random search algorithms [7, 14]. Although non-elitist EAs have been proved to optimise pseudo-Boolean functions under uncertainties [3, 4], precise parameters settings are not clear. Dang and Lehre [3] also states that a sufficiently large enough population size and a moderate mutation rate are beneficial for the non-elitist EAs to handle noise, but they do not give precise ranges of the mutation rate according to the level of uncertainties.

In this paper, we derive a general theorem (Theorem 2) for the non-elitist EA with 2-tournament selection (Algorithm 1) under

[*] Authors are listed in alphabetical order.

GECCO '21, July 10–14, 2021, Lille, France

ACM ISBN 978-1-4503-8350-9/21/07…$15.00
https://doi.org/10.1145/3449639.3459312

uncertainties. Then we analyse the runtime of Algorithm 1 on OneMax and LeadingOnes under prior and posterior noise models. In noisy settings, our analyses are more extensive and precise than previous analyses of non-elitist EAs [3]. In several settings, we prove that the non-elitist EAs beat the current state of the art results (see Tables 1-6). Note that the runtime for a certain level of noise can be obtained by plugging the noise level and the appropriate parameters into the general results in these tables. We also provide more precise guidance on how to choose mutation rate and population size as a function of the level of uncertainties. Finally, we prove for the first time that non-elitist EAs can optimise the DynBV problem in polynomial time.

2 PRELIMINARIES

In this paper, we consider a non-elitist EA with binary tournament selection optimising three noisy versions and one dynamic version of pseudo-Boolean functions. We first define some notations which are used later. For any integer $n > 0$, we define $[n] := \{1, \ldots, n\}$ and $[0..n] := \{0\} \cup [n]$. We use $\mathrm{H}(\cdot, \cdot)$ to denote Hamming-distance. The natural logarithm is denoted by $\ln(\cdot)$, and the logarithm to the base 2 is denoted by $\log(\cdot)$. Let $f : \mathcal{X} \rightarrow \mathbb{R}$ be any pseudo-Boolean function, where $\mathcal{X} = \{0, 1\}^n$ is the set of bitstrings of length n.

2.1 Uncertainties

We consider two well-known pseudo-Boolean functions OneMax and LeadingOnes which are defined as $\mathrm{OM}(x) := \sum_{i=1}^{n} x_i$ and $\mathrm{LO}(x) := \sum_{i=1}^{n} \prod_{j=1}^{i} x_j$ respectively. In this paper, we consider three noise models and the DynBV problem which are defined as follows.

For noisy optimisation, we use $f^n(x)$ instead of $f(x)$. The one-bit noise model (q) [9, 12, 13, 15, 21, 23] is the easiest starting point for theoretical analysis, which can be described as: given a probability $q \in [0, 1]$ and a solution $x \in \{0, 1\}^n$, then

$$f^n(x) = \begin{cases} f(x) & \text{with probability } 1 - q \\ f(x') & \text{with probability } q \end{cases}$$

where x' is a uniformly sampled Hamming neighbour of x.

In real-world problems, the noise can affect a stochastic number of bits rather than at most one bit. The bit-wise model (p) [13, 21, 23] can more closely imitate reality: given a probability $p \in [0, 1]$ and a solution $x \in \{0, 1\}^n$, then $f^n(x) = f(x')$ where x' independently flips each bit of the original solution x with probability p.

The Gaussian noise model (σ^2) [3, 10, 13, 22] is a type of additive noise, which adds a value independently sampled from a normal distribution in each evaluation. We can define it as: given a number $\sigma \geq 0$ and a solution $x \in \{0, 1\}^n$, then $f^n(x) = f(x) + \mathcal{N}(0, \sigma^2)$.

For dynamic optimisation, we use $f^t(x)$ instead of $f(x)$, where $t \in \mathbb{N}$ represents the current generation. Unlike noisy optimisation, the uncertainty of dynamic optimisation is reflected in different generations rather than in different evaluations. For example, the weight of each bit position is sampled from a distribution in each generation and each individual is evaluated by computing weighted sum in the random linear function [18]. The DynBV problem is a special case of the random linear function [18], which was first proposed by Lengler and Meier [16]. It can be regarded as a dynamic version of the BinVal problem. For the DynBV problem, we uniformly sample a new permutation $\pi_t : [n] \rightarrow [n]$ and evaluate the individuals in the t-th generation by $f^t(x) = \sum_{i=1}^{n} 2^{n-i} x_{\pi_t(i)}$.

Algorithm 1 Non-elitist EA with 2-tournament selection

```
Require: Fitness function $f : \mathcal{X} \rightarrow \mathbb{R}$; Population sizes $\lambda \in \mathbb{N}$
      where $\lambda \geq 2$; Mutation parameter $\chi \in (0, n)$; Initial population
      $P_0 \in \mathcal{X}^\lambda$.
 1: for $t = 0, 1, 2, \ldots$ until termination condition met do
 2:     for $i = 1$ to $\lambda$ do
 3:         $x_1 \leftarrow P_t(i_1)$ where $i_1 \backsim \mathrm{Uniform}([\lambda])$
 4:         $x_2 \leftarrow P_t(i_2)$ where $i_2 \backsim \mathrm{Uniform}([\lambda])$
 5:         if $f(x_1) \geq f(x_2)$ then $z \leftarrow x_1$ else $z \leftarrow x_2$
 6:         $P_{t+1}(i) \leftarrow y$
              ($y$ created by mutating $z$ with mutation rate $\chi/n$).
 7:     end for
 8: end for
```

2.2 Non-elitist EA with 2-tournament selection

The non-elitist EA studied in this paper is shown in Algorithm 1. In the t-th generation, we define the population $P_t \in \mathcal{X}^\lambda$, where $\lambda \geq 2$ is the population size. Algorithm 1 adopts the binary tournament selection mechanism which uniformly selects two individuals x_1 and x_2, then mutates the fittest one with mutation rate χ/n and adds it into the next population P_{t+1}. For noisy optimisation, we compare the pair of individuals based on the noisy function $f^n(x)$ instead of $f(x)$. Note that the reevaluation strategy [3, 12, 13, 21, 22] is applied in this study which means the noisy fitness value of an individual will be reevaluated though it might have been evaluated before in this generation. Similarly, for the dynamic optimisation, we replace $f(x_1) \geq f(x_2)$ with $f^t(x_1) \geq f^t(x_2)$ in line 5 of Algorithm 1.

2.3 Level-based analysis

The level-based theorem [2] is a runtime analysis tool used to obtain upper bounds on the runtime of many non-elitist EAs on a wide variety of optimisation problems [2, 5, 8]. The theorem applies to algorithms that follow the scheme of Algorithm 2. The search space $\mathcal{X}$ is partitioned into $m + 1$ disjoint subsets $A_0, A_1, \ldots, A_m$, which are called levels. The final level A_m consists of the optimal solution of function f. We denote $A_{\geq j} := \cup_{k=j}^{m} A_k$ which means at least level j. In this theorem, we consider that each individual is sampled from a distribution rather than consider genetic operators, such as selection and mutation. For example, line 3 of Algorithm 2 is responding to lines 3-6 of Algorithm 1.

Algorithm 2 Population-based Algorithm

```
Require: Finite state space $\mathcal{X}$ and population size $\lambda \in \mathbb{N}$; Map $D$
      from $\mathcal{X}^\lambda$ to the space of probability distribution over $\mathcal{X}$
Require: Initial population $P_0 \in \mathcal{X}^\lambda$
 1: for $t = 0, 1, 2, \ldots$ until termination condition met do
 2:     for $i = 1$ to $\lambda$ do
 3:         Sample $P_{t+1}(i) \sim D(P_t)$
 4:     end for
 5: end for
```

Table 1: Theoretical results of evolutionary algorithms on OneMax in the one-bit noise model (q)

Algorithm	**Parameters setting**	**Noise level q**	**Runtime**
(1+1) EA	$\chi/n = 1/n$	$O(1/n)$	$\Theta(n\log(n))$ [13]
	$\chi/n = 1/n$	$O(\log(n)/n)$	poly(n) [13]
	$\chi/n = 1/n$	$\omega(\log(n)/n)$	$2^{\omega(\log(n))}$ [13]
(1+1) EA (resampling)	$\chi/n = 1/n$; $m = 4n^3$	$[0,1]$	poly(n) [21]
(μ+1) EA	$\chi/n = 1/n$; $\mu > 12\log(15n)/q$	$(0,1]$	$O(\mu n\log(n))$ [13]
(1+λ) EA	$\chi/n = 1/n$; $\lambda \geq \max\{12/p, 24\}n\log n$	$(0,1]$	$O(n^2\lambda)$ [13]
ACO-fp	$\rho = \mathrm{o}\left(1/\left(n^3\log n\right)\right)$	$[0,1]$	$O\left(n^2\log(n)/\rho\right)$ [12]
Non-elitist EA using tournament selection (k = 2)	$\chi/n \in (0, \ln(1+2\theta\zeta)/n)$; $\lambda \in \Omega(\log(n/\chi))$ $\theta = 1/2 - (q/2)(1-q/2) - q/2n_0$, for any constant $\zeta \in (0,1)$, $n_0 \geq 3$;	$[0,1]$	$O\left(\lambda n\log(1/\chi) + n\log(n)/\chi\right)$ (Theorem 4)

Table 2: Theoretical results of evolutionary algorithms on LeadingOnes in the one-bit noise model (q)

Algorithm	**Parameters setting**	**Noise level q**	**Runtime**
(1+1) EA	$\chi/n = 1/n$	$[0,1/2]$	$\Theta\left(n^2\right)e^{\Theta\left(\min\{qn^2,n\}\right)}$ [23]
(1+1) EA (resampling)	$\chi/n = 1/n$; $m = 4n^4\log(n)/15$	$[0,1]$	$O(mn^8)$ [21]
(1+λ) EA	$\chi/n = 1/n$; $\lambda \geq 3.42\log(n)$, $\lambda \in O(n)$	$[0,1/2]$	$O\left(n^2 e^{O(n/\lambda)}\right)$ [23]
UMDA	$\mu \geq c\log(n)$; $\lambda > 4e(1+\delta)\mu$ for constants c, δ;	$[0,1)$	$O(n\lambda\log(\lambda) + n^2)$ [15]
Non-elitist EA using tournament selection (k = 2)	$\chi/n \in (0, \ln(1+2\theta\zeta)/n)$; $\lambda \in \Omega(\log(n/\chi))$ $\theta = 1/2 - q(1-q/2)$, for any constant $\zeta \in (0,1)$;	$[0,1)$	$O\left(n\lambda\log(n/\chi) + n^2/\chi\right)$ (Theorem 5)

Table 3: Theoretical results of evolutionary algorithms on OneMax in the bit-wise noise model (p)

Algorithm	**Parameters setting**	**Noise level p**	**Runtime**
(1+1) EA	$\chi/n = 1/n$	$O(1/n^2)$	$\Theta(n\log(n))$ [13]
	$\chi/n = 1/n$	$O(\log(n)/n^2)$	poly(n) [13]
	$\chi/n = 1/n$	$\omega(\log(n)/n^2)$	$2^{\omega(\log(n))}$ [13]
(1+1) EA (resampling)	$\chi/n = 1/n$; $m = n^{3+2c}/4$	$p = 1/2 - 1/n^c$ for $0 < c = \Theta(1)$	poly(n) [21]
Non-elitist EA using tournament selection (k = 2)	$\chi/n \in (0, \ln(1+2\theta\zeta)/n)$ $\theta = 9(1/2-p)/\left(64\sqrt{2pn}+16\right)$, for any constant $\zeta \in (0,1)$; $\lambda \in \Omega\left(\frac{1+pn}{(1-2p)^2}\log\left(\frac{n}{(1-2p)\chi}\right)\right)$	$(0,1/2)$	$O\left(\frac{n(1+pn)}{(1-2p)^2}\left(\lambda\log\left(\frac{1}{\chi}\right) + \frac{\log(n)}{\chi}\right)\right)$ (Theorem 6)

Table 4: Theoretical results of evolutionary algorithms on LeadingOnes in the bit-wise noise model (p)

Algorithm	**Parameters setting**	**Noise level p**	**Runtime**
(1+1) EA	$\chi/n = 1/n$	$[0, 1/(2n)]$	$\Theta\left(n^2\right)e^{\Theta\left(\min\{pn^3,n\}\right)}$ [23]
(1+1) EA (resampling)	$\chi/n = 1/n$; $m = 36n^{2c+4}$	$p = c\log(n)/n$ for $0 < c = \Theta(1)$	$12m \cdot n^{30c+1}$ [21]
	$\chi/n = 1/n$; $m \in O(\mathrm{poly}(n))$	$\omega(\log(n)/n)$	$e^{\Omega(n)}$ [21]
Non-elitist EA using tournament selection (k = 2)	$\chi/n \in (0, \ln(1+2\theta\zeta)/n)$; $\lambda \in \Omega\left(\frac{e^{6np}}{(1-3p)^2}\log\left(\frac{n}{\chi}\right)\right)$ $\theta = (1/2-p)\,e^{-3np}$, for any constant $\zeta \in (0,1)$	$[0, 1/3)$	$O\left(\frac{ne^{6np}}{(1-3p)^2}\left(\lambda\log\left(\frac{n}{\chi}\right) + \frac{n}{\chi}\right)\right)$ (Theorem 7)

Table 5: Theoretical results of evolutionary algorithms on OneMax in the Gaussian noise model (σ^2)

Algorithm	**Parameters setting**	**Noise level** σ	**Runtime**
(1+1) EA	$\chi/n = 1/n$ $\chi/n = 1/n$	$\sigma^2 \leq 1/(4\log(n))$ $\sigma^2 \geq 1$	$O(n\log(n))$ [13] $e^{\Omega(n)}$ [22]
(1+1) EA (resampling)	$\chi/n = 1/n$; $m = \lceil \frac{n\sigma^2}{\log(n)} \rceil$	$1 \leq \sigma^2 \in \text{poly}(n)$	poly(n) [22]
$(\mu + 1)$EA	$\chi/n = 1/n$; $\mu \in \text{poly}(n)$	$\sigma \geq n^3$	$2^{\omega(\log(n))}$ [11]
cGA	$K = \omega\left(\sigma^2\sqrt{n}\log(n)\right)$	$\sigma^2 > 0$	$O\left(K\sigma^2\sqrt{n}\log(Kn)\right)$ [10]
ACO-fp	$\rho = o\left(1/\left(n(n+\sigma\log n)^2\log n\right)\right)$	$\sigma^2 \geq 0$	$O\left(n^2\log(n)/\rho\right)$ [12]
Non-elitist EA using tournament selection (k = 2)	$\chi/n \in (0, \ln(1+2\theta\zeta)/n)$; $\lambda \in \Omega(\sigma^2\log(n/\chi))$ $\theta = 1/(6 + 48\sigma/\pi)$, for any constant $\zeta \in (0, 1)$;	$\sigma^2 > 0$	$O(\sigma^2\lambda n\log(1/\chi)+$ $\sigma^2 n\log(n)/\chi)$ (Theorem 8)

Table 6: Theoretical results of evolutionary algorithms on LeadingOnes in the Gaussian noise model (σ^2)

Algorithm	**Parameters setting**	**Noise level** σ	**Runtime**
(1+1) EA	$\chi/n = 1/n$ $\chi/n = 1/n$	$\sigma^2 \leq 1/\left(12en^2\right)$ $\sigma^2 \geq n^2$	$O\left(n^2\right)$ [13] $\Omega\left(e^n\right)$ [22]
(1+1) EA (resampling)	$\chi/n = 1/n$; $m = \lceil 12en^2\sigma^2 \rceil$	$n^2 \leq \sigma^2 \in \text{poly}(n)$	$O(mn^2)$ [22]
Non-elitist EA using tournament selection (k = 2)	$\chi/n \in (0, \ln(1+2\theta\zeta)/n)$; $\lambda \in \Omega(\sigma^2\log(n/\chi))$ $\theta := 1/(6 + 48\sigma/\pi)$, for any constant $\zeta \in (0, 1)$;	$\sigma^2 > 0$	$O\left(\sigma^2\lambda n\log(n/\chi) + \sigma^2 n^2/\chi\right)$ (Theorem 8)

Theorem 1 (Level-based theorem [2]). *Given a partition* $(A_0, A_1, ..., A_m)$ *of a finite state space* $\mathcal{X}$, *let* $T := \min\{t\lambda \mid |P_t \cap A_m| > 0\}$ *be the first point in time that the elements of* A_m *appear in* P_t *of Algorithm 2. If there exist* $z_0, z_1, ..., z_{m-1}, \delta \in (0, 1]$, *and* $\gamma_0 \in (0, 1)$ *such that for any population* $P \in \mathcal{X}^\lambda$,

(G1) for all $j \in [0..m-1]$, *if* $|P \cap A_{\geq j}| \geq \gamma_0\lambda$ *then*

$$\Pr_{y\sim D(P)}\left(y \in A_{\geq j+1}\right) \geq z_j,$$

(G2) for all $j \in [0..m-2]$, *and all* $\gamma \in (0, \gamma_0]$, *if* $|P \cap A_{\geq j}| \geq \gamma_0\lambda$ *and* $|P \cap A_{\geq j+1}| \geq \gamma\lambda$ *then* $\Pr_{y\sim D(P)}\left(y \in A_{\geq j+1}\right) \geq (1+\delta)\gamma$,

(G3) and the population size $\lambda \in \mathbb{N}$ *satisfies*

$$\lambda \geq 4/(\gamma_0\delta^2)\ln\left(128(m+1)/(z_*\delta^2)\right) \text{ where } z_* := \min\left\{z_j\right\},$$

then $E[T] \leq \frac{8}{\delta^2}\sum_{j=0}^{m-1}\left(\lambda\ln\left(\frac{6\delta\lambda}{4+z_j\delta\lambda}\right) + \frac{1}{z_j}\right)$.

3 MAIN RESULT

In this section, we introduce our main result (shown in Theorem 2) which is an upper bound of the expected runtime of Algorithm 1 in uncertain environments. The proof is followed.

Theorem 2. *Let* $(A_0, A_1,...,A_m)$ *be a fitness partition of a finite state space* $\mathcal{X}$. *If there exist* $h_0, h_1, ..., h_{m-1}$ *and* $\theta \in (0, 1/2]$, *and let* $T := \min\{2t\lambda \mid |P_t \cap A_m| > 0\}$ *be the first point in time that the elements of* A_m *appear in* P_t *of Algorithm 1 with noisy function* $f^n(x)$ *and mutation rate* χ/n, *where* $\chi \in (0, \ln(1+2\theta\zeta))$ *for an arbitrary constant* $\zeta \in (0, 1)$, *such that, for an arbitrary constant* $\xi \in (0, 1/16)$,

(C1) for all $j \in [0..m-1]$, $\Pr(y \in A_{\geq j+1} \mid z \in A_j) \geq h_j$,

(C2) for all $j \in [0..m-2]$, *and all search points* $x_1 \in A_{\geq j+1}$ *and* $x_2 \in A_{\leq j}$, *it follows* $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) \geq \frac{1}{2} + \theta$,

(C3) and the population size $\lambda \in \mathbb{N}$ *satisfies*

$$\lambda > \frac{4(1+o(1))}{\theta^2\xi(1-\zeta)^4}\ln\left(\frac{128(m+1)}{\theta^2\xi(1-\zeta)^4\min\{h_j\}}\right),$$

then $E[T] < \frac{16(1+o(1))}{\theta^2\xi(1-\zeta)^2}\sum_{j=0}^{m-1}\left(\lambda\ln\left(\frac{6}{\xi(1-\zeta)^2h_j}\right) + \frac{1}{\xi(1-\zeta)^2h_j}\right)$.

Proof. We use the level-based theorem (Theorem 1) to prove Theorem 2. Firstly, we derive some inequalities which are used later. From $\theta \in (0, 1/2]$, $\zeta \in (0, 1)$ and $0 < \chi < \ln(1+2\theta\zeta)$ which are assumptions of Theorem 2, we can get

$$e^\chi < 1 + 2\theta\zeta \tag{1}$$

$$(1+2\theta) - e^\chi > 2\theta(1-\zeta). \tag{2}$$

Then we define ε and γ_0 which are used later. Let constant $\varepsilon := \left(1 + \sqrt{1 - 4\sqrt{\xi}}\right)/2$, and we know that $\varepsilon \in (1/2, 1)$ by $\xi \in (0, 1/16)$. We define $\gamma_0 := \frac{(1+2\theta)-\exp(\chi)}{2\theta}(1-\varepsilon)$. By Eq. (2), we know that

$$\gamma_0 = \frac{(1+2\theta) - e^\chi}{2\theta}(1-\varepsilon) > \frac{2\theta(1-\zeta)(1-\varepsilon)}{2\theta} = (1-\zeta)(1-\varepsilon). \tag{3}$$

We first show that condition (G2) of Theorem 1 holds. We define the *current level* to be the highest level $j \in [0..m]$ such that there are at least $\gamma_0\lambda$ individuals in level j or higher, and there are fewer than $\gamma_0\lambda$ individuals in level $j+1$ or higher. Then assume that the current level is $j \leq n-2$, which means there are at least $\gamma_0\lambda$ individuals of the population P_t in $A_{\geq j}$, and at least $\gamma\lambda$ but less than $\gamma_0\lambda$ individuals in $A_{\geq j+1}$. Let x_1 and x_2 be the individuals selected from the population P_t in lines 3 and 4, z be the solution after comparison in line 5, and y be the solution after mutating corresponding to line 6 of Algorithm 1.

Now we estimate a lower bound on the probability that the offspring y is still in $A_{\geq j+1}$. By the law of total probability,

$$\Pr(y \in A_{\geq j+1}) \geq \Pr(z \in A_{\geq j+1}) \cdot \Pr(y \in A_{\geq j+1} | z \in A_{\geq j+1}).$$

The probability of selecting an individual z which is in $A_{\geq j+1}$ via binary tournament is composed of two cases. The first case is both x_1 and x_2 which are selected in lines 3 and 4 of Algorithm 1 are in $A_{\geq j+1}$ whose probability is at least γ^2. The second case is that x_1 or x_2 is evaluated to be in $A_{\geq j+1}$, whereas the other is evaluated to be in $A_{\leq j}$. In this case, noise leads to incorrect comparison in line 5 of Algorithm 1 with some probability. Let S be the event of a successful comparison, i.e. the better individual of x_1 and x_2 is exactly selected from line 5. Hence, the second case occurs with probability $2(1-\gamma)\gamma \Pr(S)$. Then,

$$\Pr(z \in A_{\geq j+1}) \geq \gamma^2 + 2(1-\gamma)\gamma \Pr(S).$$

To estimate a lower bound for $\Pr(S)$, without loss of generality, we assume that $x_1 \in A_{\geq j+1}$ and $x_2 \in A_{\leq j}$. Then, by condition (C2),

$$\Pr(S) = \Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) \geq \frac{1}{2} + \theta.$$

To estimate a lower bound for $\Pr(y \in A_{\geq j+1} \mid z \in A_{\geq j+1})$, we only consider the case that the mutation operator does not flip any bits, then by Lemma 11 and Lemma 10,

$$\begin{aligned}
&\Pr(y \in A_{\geq j+1} \mid z \in A_{\geq j+1}) \\
&\geq \left(1 - \frac{\chi}{n}\right)^n \geq e^{-\chi}\left(1 - \frac{\chi^2}{n}\right) \geq e^{-\chi}\left(1 - \frac{2\theta}{n}\right) \text{ for all } n > 1.
\end{aligned}$$

Overall, we can get a lower bound for $\Pr(y \in A_{\geq j+1})$ by plugging in $\Pr(z \in A_{\geq j+1})$, $\Pr(y \in A_{\geq j+1} | z \in A_{\geq j+1})$ and $\Pr(S)$,

$$\begin{aligned}
&\Pr(y \in A_{\geq j+1}) \\
&> \left(\gamma^2 + 2(1-\gamma)\gamma \Pr(S)\right) e^{-\chi}\left(1 - \frac{2\theta}{n}\right) \\
&\geq \left(\gamma^2 + 2(1-\gamma)\gamma\left(\frac{1}{2} + \theta\right)\right) e^{-\chi}\left(1 - \frac{2\theta}{n}\right) \\
&\geq \gamma\left(1 + 2\theta - 2\theta\gamma_0\right) e^{-\chi}\left(1 - \frac{2\theta}{n}\right)
\end{aligned}$$

by definition of $\gamma_0 = \frac{(1+2\theta)-\exp(\chi)}{2\theta}(1-\varepsilon)$,

$$\begin{aligned}
&= \gamma\left(1 + 2\theta - \left(1 + 2\theta - e^{\chi}\right) + \left(1 + 2\theta - e^{\chi}\right)\varepsilon\right) e^{-\chi}\left(1 - \frac{2\theta}{n}\right) \\
&= \gamma\left(1 + \left(1 + 2\theta - e^{\chi}\right)\varepsilon e^{-\chi}\right)\left(1 - \frac{2\theta}{n}\right)
\end{aligned}$$

letting $\delta := (1 + (1 + 2\theta - e^{\chi})\,\varepsilon e^{-\chi})\,(1 - 2\theta/n) - 1$,

$$= \gamma(1 + \delta). \tag{4}$$

Now we prove that $\delta > 0$,

$$\delta = \left(1 + \left(1 + 2\theta - e^{\chi}\right)\varepsilon e^{-\chi}\right)\left(1 - \frac{2\theta}{n}\right) - 1$$

by Eq. (1),

$$\begin{aligned}
&> \left(1 + 2\theta e^{-\chi}\varepsilon(1-\zeta)\right)\left(1 - \frac{2\theta}{n}\right) - 1 \\
&= 2\theta e^{-\chi}\varepsilon(1-\zeta) - \frac{2\theta}{n}\left(1 + 2\theta e^{-\chi}\varepsilon\right) + \frac{4\theta^2 e^{-\chi}\varepsilon\zeta}{n} \\
&> 2\theta e^{-\chi}\varepsilon(1-\zeta) - \frac{6\theta}{n} = \theta\left(2e^{-\chi}\varepsilon(1-\zeta) - \frac{6}{n}\right)
\end{aligned}$$

by Eq. (1), we have $e^{\chi} < 1 + 2\theta\zeta < 1 + 2\theta < 2$,

$$> \theta\left(\varepsilon(1-\zeta) - \frac{6}{n}\right) = \theta\varepsilon(1-\zeta)\,(1 - o(1))\,. \tag{5}$$

Thus, we get $\delta > 0$ so condition (G2) of Theorem 1 holds from Eq. 4.

To verify condition (G1), we need to estimate the probability of increasing the level of the population. We assume there are at least $\gamma_0\lambda$ individuals in A_j where $j \in [0..m-1]$. We only consider the case that the selected individuals are both in A_j in lines 3 and 4 of Algorithm 1, and the individual increases its level after mutation,

$$\Pr(y \in A_{\geq j+1}) \geq \gamma_0^2 \Pr(y \in A_{\geq j+1} \mid z \in A_{\geq j}) \geq \gamma_0^2 h_j =: z_j.$$

Condition (G3) requires the population size to satisfy

$$\frac{4}{\gamma_0\delta^2}\ln\left(\frac{128(m+1)}{\min\{z_j\}\delta^2}\right) \leq \frac{4}{\gamma_0\delta^2}\ln\left(\frac{128(m+1)}{\gamma_0^2 \min\{h_j\}\delta^2}\right)$$

by Eq. (3) and (5),

$$\begin{aligned}
&< \frac{4\,(1+o(1))}{(1-\zeta)(1-\varepsilon)\,(\theta\varepsilon(1-\zeta))^2} \\
&\qquad \cdot \ln\left(\frac{128(m+1)\,(1+o(1))}{(1-\zeta)^2(1-\varepsilon)^2\,(\theta\varepsilon(1-\zeta))^2 \min\{h_j\}}\right) \\
&= \frac{4\,(1+o(1))}{\theta^2\varepsilon^2(1-\varepsilon)(1-\zeta)^3}\ln\left(\frac{128(m+1)}{\theta^2\varepsilon^2(1-\varepsilon)^2(1-\zeta)^4\min\{h_j\}}\right) \\
&< \frac{4\,(1+o(1))}{\theta^2\varepsilon^2(1-\varepsilon)^2(1-\zeta)^4}\ln\left(\frac{128(m+1)}{\theta^2\varepsilon^2(1-\varepsilon)^2(1-\zeta)^4\min\{h_j\}}\right)
\end{aligned}$$

because $\varepsilon^2(1-\varepsilon)^2 = \xi$ by the definition of $\varepsilon = \left(1 + \sqrt{1 - 4\sqrt{\xi}}\right)/2$,

$$< \frac{4\,(1+o(1))}{\theta^2\xi(1-\zeta)^4}\ln\left(\frac{128(m+1)}{\theta^2\xi(1-\zeta)^4\min\{h_j\}}\right) < \lambda.$$

Therefore, condition (C3) of Theorem 2 guarantees that the population size satisfies condition (G3) of Theorem 1.

Finally, all conditions of Theorem 1 hold and the expected time (the reevaluation strategy is taken into account) to reach the optimum is no more than

$$\begin{aligned}
&E[T] \\
&\leq 2 \cdot \frac{8}{\delta^2}\sum_{j=0}^{m-1}\left(\lambda\ln\left(\frac{6\delta\lambda}{4 + z_j\delta\lambda}\right) + \frac{1}{z_j}\right) < \frac{16}{\delta^2}\sum_{j=0}^{m-1}\left(\lambda\ln\left(\frac{6}{z_j}\right) + \frac{1}{z_j}\right) \\
&\leq \frac{16}{\delta^2}\sum_{j=0}^{m-1}\left(\lambda\ln\left(\frac{6}{\gamma_0^2 h_j}\right) + \frac{1}{\gamma_0^2 h_j}\right)
\end{aligned}$$

by Eq. (3) and (5),

$$
\begin{aligned}
&< \frac{16\,(1+o(1))}{\theta^2\varepsilon^2(1-\zeta)^2} \sum_{j=0}^{m-1} \left(\lambda \ln\left(\frac{6}{(1-\varepsilon)^2(1-\zeta)^2 h_j}\right) + \frac{1/h_j}{(1-\varepsilon)^2(1-\zeta)^2}\right) \\
&< \frac{16\,(1+o(1))}{\theta^2\varepsilon^2(1-\varepsilon)^2(1-\zeta)^2} \sum_{j=0}^{m-1} \left(\lambda \ln\left(\frac{6}{\varepsilon^2(1-\varepsilon)^2(1-\zeta)^2 h_j}\right) \right.\\
&\qquad \left. + \frac{1/h_j}{\varepsilon^2(1-\varepsilon)^2(1-\zeta)^2}\right) \\
&= \frac{16\,(1+o(1))}{\theta^2\xi(1-\zeta)^2} \sum_{j=0}^{m-1} \left(\lambda \ln\left(\frac{6}{\xi(1-\zeta)^2 h_j}\right) + \frac{1}{\xi(1-\zeta)^2 h_j}\right).
\end{aligned}
$$

□

4 NOISY OPTIMISATION

Now we can apply Theorem 2 to several noisy settings and derive the runtime and the parameters required. The most technical part is to estimate condition (C2) of Theorem 2, that is compute lower bounds for the probability that the real fitter individual is exactly selected in line 5 of Algorithm 1. In Lemma 3, we summarise lower bounds in some scenarios, and the proof is shown in Appendix B.1.

LEMMA 3. *Let $A_j := \{x \in \{0,1\}^n | f(x) = j\}$ for all $j \in [0..n]$ be a partition. Let x_1 and x_2 be two individuals in $A_{\geq j+1}$ and $A_{\leq j}$ respectively, where $j \in [0..n-2]$, then $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) > 1/2 + \theta$ where*

(a) $\theta = 1/2 - q/2(1 - q/2) - q/(2n_0)$ for $q \in [0,1)$ and $n_0 \in [3,\infty)$ on OneMax in the one-bit noise model (q),
(b) $\theta = 1/2 - q(1-q)$ for $q \in [0,1)$ on LeadingOnes in the one-bit noise model (q),
(c) $\theta = \frac{9(1/2-p)}{64\sqrt{2pn}+16}$ for $p \in (0,1/2)$ on OneMax in the bit-wise noise model (p),
(d) $\theta = (1/2 - 3p/2)\,e^{-3np}$ for $q \in [0,1/3)$ on LeadingOnes in the bit-wise noise model (p), and
(e) $\theta = 1/(6 + 48\sigma/\pi)$ for $\sigma > 0$ on OneMax and LeadingOnes in the Gaussian noise model (σ^2).

We also need to derive lower bounds for probability of sampling individuals beyond the current level of the population (condition (C1)) and the population size required (condition (C3)). Then we can conclude the runtimes and the mutation rates required from Theorem 2. In this section, Theorem 4 and 5 show the results in the one-bit noise model on OneMax and LeadingOnes respectively. Theorems 6 and 7 show the results in the bit-wise noise model on OneMax and LeadingOnes respectively. Theorem 8 shows the results in the Gaussian noise model on OneMax and LeadingOnes. The proofs of Theorems 4-8 are shown in Appendix B.2-B.6 respectively.

4.1 One-bit noise model

THEOREM 4. *For any constant $q \in [0,1]$, any constant $\zeta \in (0,1)$, any constant $n_0 \in [3,\infty)$ and any $\chi \in (0, \ln(1+2\theta\zeta))$, where $\theta := 1/2 - (q/2)(1-q/2) - q/(2n_0)$, Algorithm 1 with mutation rate χ/n and population size $\lambda > c\log(n/\chi)$ for a sufficiently large enough constant c achieves the optimum on OneMax in expected time $O(\lambda n \log(1/\chi) + n\log(n)/\chi)$ in the one-bit noise model (q).*

THEOREM 5. *For any constant $q \in [0,1)$ and any constant $\zeta \in (0,1)$, any $\chi \in (0, \ln(1+2\theta\zeta))$, where $\theta := 1/2 - q(1-q/2)$, Algorithm 1 with mutation rate χ/n and population size $\lambda > c\log(n/\chi)$ for a sufficiently large enough constant c achieves the optimum on LeadingOnes in expected time $O\left(n\lambda\log(n/\chi) + n^2/\chi\right)$ in the one-bit noise model (q).*

Theorems 4 and 5 imply that one-bit noise does not impact the asymptotical runtime of Algorithm 1 if we choose constant mutation parameter χ which satisfies the assumption, but we have less choices of mutation rate as the level of is noise growing. In contrast, the (1+1) EA becomes inefficient if the noise level is a constant (see Tables 1 and 2). Compared to other EAs, e.g., ACO-fp, UMDA and (1+1) EA (resampling), the non-elitist EAs beat the current state of the art results in these two settings (see Tables 1 and 2).

4.2 Bit-wise noise model

THEOREM 6. *For any $p \in (0,1/2)$, any constant $\zeta \in (0,1)$ and any $\chi \in (0, \ln(1+2\theta\zeta))$, where $\theta := 9(1/2-p)/\left(64\sqrt{2pn}+16\right)$, Algorithm 1 with mutation rate χ/n and population size $\lambda > \frac{c(1+pn)}{(1-2p)^2}\log\left(\frac{n}{(1-2p)\chi}\right)$ for a sufficiently large enough constant c achieves the optimum on OneMax in expected time $O\left(\frac{n(1+pn)}{(1-2p)^2}\left(\lambda\log\left(\frac{1}{\chi}\right) + \frac{\log(n)}{\chi}\right)\right)$ in the bit-wise noise model (p).*

By Theorem 6, we can compute that for extremely high-levels of bit-wise noise, i.e. $p = 1/2 - 1/n^b$ where constant $b > 0$, Algorithm 1 with mutation rate $\chi/n = \theta\zeta/n$ which is less than $\ln(1+2\theta\zeta)/n$ by Lemma 15, i.e., $\chi = d/n^{b+1/2}$ for some constant $d > 0$, and a sufficiently large enough population size $\lambda \in \Omega\left(n^{2b+1}\log(n)\right)$ has polynomial expected runtime $O\left(n^{2b+2}\lambda\log(n)\right)$ on OneMax.

THEOREM 7. *For any $p \in [0,1/3)$, any constant $\zeta \in (0,1)$ and any $\chi \in (0, \ln(1+2\theta\zeta))$, where $\theta := (1/2 - 3p/2)\,e^{-3np}$, Algorithm 1 with mutation rate χ/n and population size $\lambda \geq c\frac{e^{6np}}{(1-3p)^2}\log\left(\frac{n}{\chi}\right)$ for a sufficiently large enough constant c achieves the optimum on LeadingOnes in expected time $O\left(\frac{ne^{6np}}{(1-3p)^2}\left(\lambda\log\left(\frac{n}{\chi}\right) + \frac{n}{\chi}\right)\right)$ in the bit-wise noise model (p).*

Now we consider the case of the extremely high bit-wise noise $p = b\log(n)/n$ for any constant $b > 0$. By Theorem 7, we can get that Algorithm 1 with mutation rate $\chi/n = \theta\zeta/n$ which satisfies the condition, i.e., $\chi = d/n^{3b}$ for some constant $d > 0$, and a sufficiently large enough population size $\lambda \in \Omega\left(n^{6b}\log(n)\right)$ achieves the optimum on LeadingOnes in expected time $O\left(n^{6b+1}\lambda\log(n) + n^{9b+2}\right)$.

4.3 Gaussian noise model

THEOREM 8. *For any $\sigma > 0$, any constant $\zeta \in (0,1)$ and any $\chi \in (0, \ln(1+2\theta\zeta))$, where $\theta := 1/(6+48\sigma/\pi)$, Algorithm 1 with mutation rate χ/n and population size $\lambda > c\sigma^2\log(n/\chi)$ for a sufficiently large enough constant c achieves the optimum on OneMax and LeadingOnes in expected time $O\left(\sigma^2\lambda n\log(1/\chi) + \sigma^2 n\log(n)/\chi\right)$ and $O\left(\sigma^2\lambda n\log(n/\chi) + \sigma^2 n^2/\chi\right)$ respectively in the Gaussian noise model (σ^2).*

Theorem 8 implies that Algorithm 1 with mutation rate $\chi/n = \theta\zeta/n$, i.e., $\chi = d/\sigma$ for some constant $d > 0$, and a sufficiently large enough population size $\lambda \in \Omega\left(\sigma^2 \log(\sigma n)\right)$ can optimise OneMax and LeadingOnes in polynomial runtime even if $\sigma^2 \in \mathrm{poly}(n)$. Similarly to the optimisation in the bit-wise noise model, the mutation rate should be fairly conservative and the population size should be large enough if the noise level is extremely high, e.g., $\sigma = n^b$ where constant $b > 0$.

5 DYNAMIC OPTIMISATION

Now we consider dynamic optimisation. We apply the main theorem (Theorem 2) on the DynBV problem and derive the runtime and the parameters required. The proof idea is similar to noisy optimisation.

Theorem 9. *For any constant $\zeta \in (0,1)$ and any $\chi \in (0, \ln(1+2\theta\zeta))$, where $\theta := 1/(2n)$, Algorithm 1 with mutation rate χ/n and population size $\lambda > cn^2\log(n)$ for a large enough constant c achieves the optimum on DynBV in expected time $O\left(n^3\lambda\log(1/\chi) + n^3\log(n)/\chi\right)$.*

Proof. We apply Theorem 2 to prove Theorem 9. We first partition the search space into levels. We use the partition $A_j := \{\, x \in \{0,1\}^n \mid \mathrm{OM}(x) = j \,\}$ for $j \in [0..n]$. It is easy to see that θ satisfies the assumption in Theorem 2.

We first show that condition (C2) of Theorem 2 holds. Let x_1 and x_2 be two individuals in $A_{\geq j+1}$ and $A_{\leq j}$ respectively, where $j \in [0..n-2]$. Let E be the event that $f^t(x_1) > f^t(x_2)$ or individual x_1 is selected uniformly from $\{x_1, x_2\}$ if $f^t(x_1) = f^t(x_2)$. The probability of this event is $\Pr(f^t(x_1) > f^t(x_2)) + \frac{1}{2}\Pr(f^t(x_1) = f^t(x_2)) = \Pr(E)$.

To estimate a lower bound for $\Pr(E)$ on DynBV, we pessimistically assume that $x_1 \in A_{j+1}$ and $x_2 \in A_{\leq j}$, such that $\mathrm{OM}(x_1) = \mathrm{OM}(x_2) + h$ where $h \in [1, j]$. We assume $\mathrm{H}(x_1, x_2) \leq l + l + h = s$, where $s \leq n$ and there exist l bit-positions that x_1 is with 1-bit and x_2 is with 0-bit, and there exist another l bit-positions that x_1 is with 0-bit and x_2 is with 1-bit, such that x_1 and x_2 have the same bit in the rest of $n - 2l - h$ position. For the DynBV problem, the coefficients vary exponentially, thus the largest coefficient is the deciding factor for the fitness value. We first compare the fitness in the $n - 2l - h$ positions of x_1 and x_2, which are the same in the same generation. The next largest coefficient decides the final fitness value. Then we say that x_1 "wins" if the event E happens. If the next largest coefficient is in the $l + h$ positions, x_1 wins, else in another l positions, x_2 wins. Therefore,

$$\Pr(E) \geq \frac{l+h}{2l+h} = \frac{l + h/2 + h/2}{2\,(l + h/2)} = \frac{1}{2} + \frac{h}{2(2l+h)}$$
$$\geq \frac{1}{2} + \frac{1}{2(2l+h)} = \frac{1}{2} + \frac{1}{2s}$$

since the Hamming-distance s between any pair of individuals is at most n, then

$$\geq \frac{1}{2} + \theta.$$

The condition (C2) of Theorem 2 holds. Since $s \leq n$, we get $\theta \geq 1/(2n)$.

To verify condition (C1), we need to estimate the probability of sampling individuals beyond the current level of the population. We assume that there is an individual $z \in A_j$ where $j \in [0..n-1]$, and let y be obtained from z by the mutation operator with mutation rate χ/n. For a lower bound, it suffices to only consider the case that none of the 1-bits are flipped and one of 0-bits is flipped after mutation. Then, by Lemma 13 it follows,

$$\Pr(y \in A_{\geq j+1} \mid z \in A_j) > \left(1 - \frac{\chi}{n}\right)^j \frac{\chi}{n}(n-j)$$
$$\geq e^{-\chi}\,(1 - o(1))\,(n-j)\chi/n =: h_j$$
$$= \Omega\left(\frac{(n-j)\chi}{n}\right).$$

Then we compute the population size required by condition (C3). Let $\xi \in (0, 1/16)$ be a constant, then

$$\lambda > \frac{4\,(1+o(1))}{\theta^2\xi(1-\zeta)^4}\ln\left(\frac{128(m+1)}{\theta^2\xi(1-\zeta)^4\min\{h_j\}}\right) = O\left(n^2\log(n/\chi)\right).$$

Condition (C3) is satisfied by $\lambda > cn^2\log(n/\chi)$ for a sufficiently large constant c.

Finally, all conditions of Theorem 2 hold and the expected time $E[T]$ to reach the optimum is no more than

$$E[T] \leq \frac{16\,(1+o(1))}{\theta^2\xi(1-\zeta)^2}\left(\lambda\sum_{j=0}^{m-1}\ln\left(\frac{6}{\xi(1-\zeta)^2 h_j}\right) + \frac{1}{\xi(1-\zeta)^2}\sum_{j=0}^{m-1}\frac{1}{h_j}\right)$$
$$= O\left(s^2\left(\lambda\sum_{j=0}^{m-1}\ln\left(\frac{n}{(n-j)\chi}\right) + \sum_{j=0}^{m-1}\frac{n}{(n-j)\chi}\right)\right)$$
$$= O\left(s^2\left(\lambda\ln\left(\frac{n^n}{\chi^n n!}\right) + n\log(n)/\chi\right)\right)$$

using the bounds $n! > (n/e)^n$, and $s \leq n$,

$$= O\left(s^2 n\lambda\log(1/\chi) + s^2 n\log(n)/\chi\right)$$
$$= O\left(n^3\lambda\log(1/\chi) + n^3\log(n)/\chi\right).$$

□

Although Lengler and Schaller [18] proved that the (1+1) EA can achieve the optimum in $O(n\log(n))$ with standard mutation rate $\chi/n = 1/n$ on the noisy linear function, there only exists a partial result for population-based EAs, i.e., runtime when the population is initiated close to the optimum [16]. Theorem 9 gives for the first time the runtime from any start point on DynBV for a population-based EA. It implies that if choosing a low mutation rate, e.g., $\chi/n = \zeta/(2n^2)$ and a population size $\lambda > cn^2\log(n)$ for a sufficiently large constant c, Algorithm 1 can optimise the DynBV problem in $O\left(n^3\lambda\log(n)\right)$ time. The analysis could be further improved by estimating the maximal Hamming-distance in all pairs of individuals more precisely.

6 EXPERIMENTAL RESULTS

In the previous sections, our main result indicates an upper bound of mutation rate to guarantee polynomial runtime as a function of the level of uncertainties. However, the current analyses do not show what happens if the mutation rate goes higher in uncertain environments. We thus conduct an experiment to complement the theoretical results. We run Algorithm 1 with different mutation parameters χ from 0.05 to 1.0 with step 0.05 on LeadingOnes under different levels of one-bit noise q from 0.0 to 1.0 with step

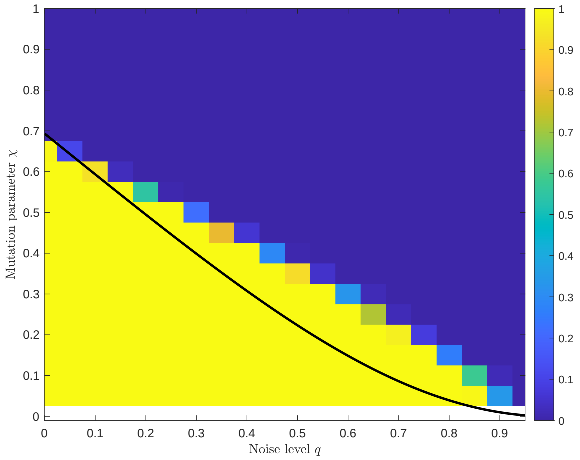

Figure 1: Success rate of 100 runs of Algorithm 1 with mutation rate $\chi/n = \chi/200$ and population size $\lambda = 30\ln(200)$ on LeadingOnes ($n = 200$) in the one-bit noise model (q) with a budget of 200^3 evaluation times. The black line indicates an upper bound of mutation parameter χ which leads to an $O(n^2)$ runtime according to noise level q.

0.05. For each pair of χ and q, we run 100 times and record the frequency of the algorithm finding the optimum with a budget of n^3 fitness evaluations. It is a time-consuming experiment, so we run the experiments with problem size $n = 200$ and show the results by a heat-map (Figure 1). From Theorem 5, we can compute the maximal value of χ to guarantee $O(n^2)$ runtime according to the level of one-bit noise (the black line in Figure 1). We can see from Figure 1 that the algorithm can successfully find the optimum in 200^3 fitness evaluation if we choose χ below the black line, which match the theoretical results. If we choose χ a bit higher than the black line, the algorithm may no longer optimise LeadingOnes in 200^3 fitness evaluations.

7 CONCLUSION

In this paper, we improve the results from Dang and Lehre [3] and extend them to more uncertainty models. We also provide more precise guidance on how to choose mutation rate and population size as a function of the level of uncertainties. From Tables 1-6, we can conclude that by using an appropriate mutation parameter, i.e., $\chi \in [\theta\zeta, \ln(1 + 2\theta\zeta))$ where $\zeta \in (0, 1)$ is a arbitrary constant and θ is a function of the level of uncertainty, and a sufficiently large enough population size, the non-elitist EA with 2-tournament selection can guarantee less time to optimise OneMax and the LeadingOnes under one-bit and extremely high-level bit-wise noise, compared to (1+1) EA using resampling strategy [21, 22]. In some settings, such as in the Gaussian noise model, we obtain a lower upper bound of runtimes than ACO-fp [12], and a comparable upper bound with EDAs [10, 15]. Finally, we prove for the first time that with appropriate parameters settings, non-elitist EAs can optimise the DynBV problem in expected polynomial time.

Future work should investigate what value of mutation rate is too high for optimisation under uncertainty. Although we can determine the appropriate mutation rate for a given uncertainty level, the noise level is often unknown in real-world optimisation. Thus, another future work would be to investigate the performance of mutation rate adjusting mechanisms, e.g., self-adaptation [1] and self-adjusting [6], under uncertainties.

ACKNOWLEDGMENTS

This work was supported by a Turing AI Fellowship (EPSRC grant ref EP/V025562/1).

REFERENCES

[1] Brendan Case and Per Kristian Lehre. 2020. Self-adaptation in non-Elitist Evolutionary Algorithms on Discrete Problems with Unknown Structure. *IEEE Transactions on Evolutionary Computation* (2020), 1–1.

[2] Dogan Corus, Duc-Cuong Dang, Anton V. Eremeev, and Per Kristian Lehre. 2018. Level-Based Analysis of Genetic Algorithms and Other Search Processes. *IEEE Transactions on Evolutionary Computation* 22, 5 (Oct. 2018), 707–719.

[3] Duc-Cuong Dang and Per Kristian Lehre. 2015. Efficient Optimisation of Noisy Fitness Functions with Population-based Evolutionary Algorithms. In *Proceedings of the 2015 ACM Conference on Foundations of Genetic Algorithms XIII - FOGA '15*. ACM Press, Aberystwyth, United Kingdom, 62–68.

[4] Duc-Cuong Dang and Per Kristian Lehre. 2016. Runtime Analysis of Non-elitist Populations: From Classical Optimisation to Partial Information. *Algorithmica* 75, 3 (July 2016), 428–461.

[5] Duc-Cuong Dang, Per Kristian Lehre, and Phan Trung Hai Nguyen. 2019. Level-Based Analysis of the Univariate Marginal Distribution Algorithm. *Algorithmica* 81, 2 (Feb. 2019), 668–702.

[6] Benjamin Doerr, Carola Doerr, and Johannes Lengler. 2019. Self-adjusting mutation rates with provably optimal success rules. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, Prague Czech Republic, 1479–1487.

[7] Benjamin Doerr, Thomas Jansen, Dirk Sudholt, Carola Winzen, and Christine Zarges. 2013. Mutation Rate Matters Even When Optimizing Monotonic Functions. *Evolutionary Computation* 21, 1 (March 2013), 1–27.

[8] Benjamin Doerr and Timo Kötzing. 2020. Multiplicative Up-Drift. *Algorithmica* (Oct. 2020).

[9] Stefan Droste. 2004. Analysis of the (1 + 1) EA for a Noisy OneMax. In *Genetic and Evolutionary Computation – GECCO 2004*. Vol. 3102. Springer Berlin Heidelberg, Berlin, Heidelberg, 1088–1099.

[10] Tobias Friedrich, Timo Kotzing, Martin S. Krejca, and Andrew M. Sutton. 2016. The Compact Genetic Algorithm is Efficient under Extreme Gaussian Noise. *IEEE Transactions on Evolutionary Computation* (2016), 1–1.

[11] Tobias Friedrich, Timo Kötzing, Martin Krejca, and Andrew M. Sutton. 2015. The Benefit of Sex in Noisy Evolutionary Search. In *Algorithms and Computation - 26th International Symposium, ISAAC 2015, Nagoya, Japan, December 9-11, 2015, Proceedings*, Vol. 9472. Springer, 140–150.

[12] Tobias Friedrich, Timo Kötzing, Martin S. Krejca, and Andrew M. Sutton. 2016. Robustness of Ant Colony Optimization to Noise. *Evolutionary Computation* 24, 2 (2016), 237–254. Publisher: MIT Press.

[13] Christian Gießen and Timo Kötzing. 2016. Robustness of Populations in Stochastic Environments. *Algorithmica* 75, 3 (July 2016), 462–489.

[14] Per Kristian Lehre. 2010. Negative Drift in Populations. In *Parallel Problem Solving from Nature, PPSN XI*, Robert Schaefer, Carlos Cotta, Joanna Kołodziej, and Günter Rudolph (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 244–253.

[15] Per Kristian Lehre and Phan Trung Hai Nguyen. 2019. Runtime analysis of the univariate marginal distribution algorithm under low selective pressure and prior noise. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, Prague Czech Republic, 1497–1505.

[16] Johannes Lengler and Jonas Meier. 2020. Large Population Sizes and Crossover Help in Dynamic Environments. In *Parallel Problem Solving from Nature – PPSN XVI*. Springer International Publishing, Cham, 610–622.

[17] Johannes Lengler and Simone Riedi. 2020. Runtime analysis of the (mu+1)-EA on the Dynamic BinVal function. *arXiv:2010.13428 [cs]* (Oct. 2020). arXiv: 2010.13428.

[18] Johannes Lengler and Ulysse Schaller. 2018. The (1+1)-EA on Noisy Linear Functions with Random Positive Weights. In *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*. 712–719.

[19] Constantin P. Niculescu and Andrei Vernescu. 2004. A two sided estimate of e^x-(1+x/n)^n. *Journal of Inequalities in Pure and Applied Mathematics* 5, 3 (2004).

[20] Adam Prugel-Bennett, Jonathan Rowe, and Jonathan Shapiro. 2015. Run-Time Analysis of Population-Based Evolutionary Algorithm in Noisy Environments. In *Proceedings of the 2015 ACM Conference on Foundations of Genetic Algorithms XIII - FOGA '15*. ACM Press, Aberystwyth, United Kingdom, 69–75.

[21] Chao Qian, Chao Bian, Wu Jiang, and Ke Tang. 2019. Running Time Analysis of the (1+1)-EA for OneMax and LeadingOnes Under Bit-Wise Noise. *Algorithmica* 81, 2 (Feb. 2019), 749–795.

[22] Chao Qian, Yang Yu, Ke Tang, Yaochu Jin, Xin Yao, and Zhi-Hua Zhou. 2018. On the Effectiveness of Sampling for Evolutionary Optimization in Noisy Environments. *Evolutionary computation* 26 (2018), 237–267.

[23] Dirk Sudholt. 2020. Analysing the Robustness of Evolutionary Algorithms to Noise: Refined Runtime Bounds and an Example Where Noise is Beneficial. *Algorithmica* (Jan. 2020).

[24] Flemming Topsoe. 2007. Some bounds for the logarithmic function. *Inequality Theory and Applications* 4 (Jan. 2007), 137.

[25] Yaochu Jin and Jürgen Branke. 2005. Evolutionary optimization in uncertain environments-a survey. *IEEE Transactions on Evolutionary Computation* 9, 3 (June 2005), 303–317.

A USEFUL INEQUALITIES

Lemma 10. *$\ln(1+x) < \sqrt{x}$ for $0 \le x < \infty$.*

Proof. By Eq. (14) in [24],

$$\ln(1+x) \le \frac{x}{\sqrt{1+x}} < \frac{x}{\sqrt{x}} = \sqrt{x}.$$

□

Lemma 11 ([19]). *$\left(1+\frac{x}{n}\right)^n \ge e^x\left(1-\frac{x^2}{n}\right)$ for $n \in \mathbb{N}^*$,$|x|\le n$.*

Lemma 12. *$\left((1-x)^{1/x-1}\right)^y \ge e^{-y}$ for $0 < x < 1$ and $y > 0$.*

Proof. By Lemma 11,

$$(1-x)^{1/x} \ge e^{-1}(1-x) \tag{6}$$

$$(1-x)^{1/x-1} \ge e^{-1} \tag{7}$$

$$\left((1-x)^{1/x-1}\right)^y \ge e^{-y} \tag{8}$$

□

Lemma 13. *$\left(1-\frac{\chi}{n}\right)^i \frac{\chi}{n} \ge e^{-\chi}(1-o(1))\frac{\chi}{n}$ for $0 < \chi = O(1)$, $n \in \mathbb{N}^*$ and $0 \le i \le n$.*

Proof.

$$\left(1-\frac{\chi}{n}\right)^i \frac{\chi}{n} \ge \left(1-\frac{\chi}{n}\right)^n \frac{\chi}{n}$$

by Lemma 11,

$$\ge e^{-\chi}\left(1-\frac{\chi^2}{n}\right)\frac{\chi}{n} = e^{-\chi}(1-o(1))\,\chi/n$$

□

Lemma 14 ([3]). *Let $F(x)$ be the cumulative density function of normal distribution $\mathcal{N}\left(0,\sigma^2\right)$, for $x > 0$ we have*

$$F(x) > 1 - \frac{1}{\sqrt{\frac{x\pi}{\sigma\sqrt{2}}+4}}$$

Lemma 15. *For any $\theta \in (0,1/2]$ and any constant $\zeta \in (0,1)$, $\theta\zeta < \ln(1+2\theta\zeta)$.*

Proof. By $\frac{2x}{2+x} \le \ln(1+x)$ from Eq. 3 in [24], we can get

$$\ln(1+2\theta\zeta) \ge \frac{4\theta\zeta}{2+2\theta\zeta} = \frac{2\theta\zeta}{1+\theta\zeta} > \frac{2\theta\zeta}{2} = \theta\zeta.$$

□

Lemma 16 ([4]). *Let X and Y be identically distributed independent random variables with integer support, finite expected value μ and finite non-zero variance σ^2, it holds that*

$$\Pr(X=Y) \ge \frac{\left(1-1/d^2\right)^2}{2d\sigma+1} \text{ for any } d \ge 1$$

B PROOFS

B.1 Proof of Lemma 3

Proof. Let E be the event that $f^n(x_1) > f^n(x_2)$ or individual x_1 is selected uniformly from $\{x_1,x_2\}$ if $f^n(x_1) = f^n(x_2)$, then $\Pr(E) = \Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2))$. Now we derive θ in different cases.

(a) To estimate a lower bound for $\Pr(E)$ on the OneMax problem in the one-bit noise model (q), we pessimistically assume that $x_1 \in A_{j+1}$ and $x_2 \in A_j$. Then we say that x_1 "wins" if the event E happens, and we distinguish four cases:

- Let E_{00} be the event that there is no noise, and x_1 wins, then $\Pr(E_{00}) = (1-q)^2$.
- Let E_{01} be the event that there is no noise in x_1 and noise in x_2, and x_1 wins, then $\Pr(E_{01}) \ge (1-q)q\left(\left(1-\frac{j}{n}\right)\frac{1}{2}+\frac{j}{n}\right) = q(1-q)\left(\frac{j}{2n}+\frac{1}{2}\right)$.
- Let E_{10} be the event that there is noise in x_1 and no noise in x_2, and x_1 wins, then $\Pr(E_{10}) \ge q(1-q)\left(\frac{j+1}{n}\cdot\frac{1}{2}+\left(1-\frac{j+1}{n}\right)\right) = q(1-q)\left(-\frac{j}{2n}-\frac{1}{2n}+1\right)$.
- Let E_{11} be the event that there is noise in x_1 and x_2, and x_1 wins. There are two situations leading x_1 to win:

  (1) The noise flips one of $j+1$ 1-bits of x_1 and one of j 1-bit of x_2.

  (2) The noise flips one of $n-(j+1)$ 0-bits of x_1.

  Thus,

$$\Pr(E_{11}) \ge q^2\left(\frac{j+1}{n}\cdot\frac{j}{n}+\left(1-\frac{j+1}{n}\right)\right) = \left(\frac{(j+1)(j-n)}{n^2}+1\right)q^2$$

since $(j+1)(j-n)$ achieves the minimum when $j = n/2$,

$$\ge \left(\frac{3}{4}-\frac{1}{2n}\right)q^2.$$

By combining all four cases above, we can get

$$\begin{aligned}
\Pr(E) &\ge \Pr(E_{00}) + \Pr(E_{01}) + \Pr(E_{10}) + \Pr(E_{11}) \\
&\ge (1-q)^2 + q(1-q)\left(\frac{j}{2n}+\frac{1}{2}\right) \\
&\quad + q(1-q)\left(-\frac{j}{2n}-\frac{1}{2n}+1\right) + \left(\frac{3}{4}-\frac{1}{2n}\right)q^2 \\
&= 1-\frac{q}{2}+\frac{q^2}{4}-\frac{q}{2n} = \frac{1}{2}+\frac{1}{2}-\frac{q}{2}\left(1-\frac{q}{2}\right)-\frac{q}{2n} \\
&\ge \frac{1}{2}+\frac{1}{2}-\frac{q}{2}\left(1-\frac{q}{2}\right)-\frac{q}{2n_0} = \frac{1}{2}+\theta
\end{aligned}$$

(b) To estimate a lower bound for $\Pr(E)$ on the LeadingOnes problem in the one-bit noise model (q), we pessimistically assume that $x_1 \in A_{j+1}$ and $x_2 \in A_j$. We also pessimistically assume that the suffix of x_1, i.e. the bits after the $(j+2)$-th position, is all 0-bits, and the suffix of x_2, i.e. the bits after the $(j+1)$-th position, is all 1-bits, which is the worst case because if the noise flips the $(j+1)$-th bit in x_2, then x_2 will have the maximal noisy fitness n. We say that x_1 "wins" if the event E happens, then we distinguish four cases to estimate $\Pr(E)$:

- Let E_{00} be the event that there is no noise, and x_1 wins, then $\Pr(E_{00}) = (1-q)^2 = q^2-2q+1$.

- Let E_{01} be the event that there is no noise in x_1 and noise in x_2, and x_1 wins. By the assumption of x_2, x_1 only fails if noise flips the only 0-bit in x_2. Thus, $\Pr(E_{01}) \geq (1-q) \cdot q \cdot (1-1/n) = -(1-1/n)\,q^2 + (1-1/n)\,q$.
- Let E_{10} be the event that there is noise in x_1 and no noise in x_2, and x_1 wins. Unless any of the first $j+1$ 1-bits of x_1 is flipped, x_1 wins. Therefore, $\Pr(E_{10}) \geq q \cdot (1-q) \cdot (1-(j+1)/n) = -(1-(j+1)/n) \cdot q^2 + (1-(j+1)/n)\,q$.
- Let E_{11} be the event that there is noise in x_1 and x_2, and x_1 wins. Because $j = n-2$ is a special case, we first estimate the probability $\Pr(E_{11})$ when $j \leq n-3$. There are four situations leading x_1 to win:
  (1) The noise does not flip the first $j+1$ 1-bits of x_1, and does not flip the 0-bit of x_2.
  (2) The noise flips the i-th 1-bits of x_1 where $i \leq j+1$, and flips one of the first $i-1$ 1-bits of x_2,
  (3) the noise flips the same bit-position in the first j 1-bits of x_1 and x_2 (tie and half chance to win).
  (4) The noise flips the $(j+1)$-th 1-bit of x_1, and does not flip the first j 1-bits of x_2 (tie and half chance to win).

  Thus,

$$\Pr(E_{11}) \geq q^2\left(\left(1-\frac{j+1}{n}\right)\left(1-\frac{1}{n}\right)+\sum_{i=2}^{j+1}\left(\frac{i-1}{n}\cdot\frac{1}{n}\right)+\frac{j}{2n^2}\right.$$
$$\left.+\frac{1}{2n}\left(1-\frac{j+1}{n}\right)\right)$$
$$=\left(j^2/2-(n-3/2)\,j+(n-1)(2n-1)/2\right)\frac{q^2}{n^2}$$

  since $1/2j^2 - (n-3/2)j + (n-1)(2n-1)/2$ is monotone decreasing as j increasing when $j \leq n-3/2$, $\Pr(E_{11})$ achieves the minimum if $j = n-3$,

$$\geq \frac{1}{2}\left(1+\frac{1}{n^2}\right)q^2 > \frac{q^2}{2}.$$

  Then we estimate $\Pr(E_{11})$ in the special case $j = n-2$. Since both x_1 and x_2 have one 0-bit to the optimum, i.e. x_1 has only one 0-bit in the last position and x_2 has only one 0-bit in the penultimate position, there are five situations to leading x_1 to win:
  (1) The noise flips the i-th 1-bits of x_1 where $i \leq n-1$, and flips one of the first $i-1$ 1-bits.
  (2) The noise flips the same bit-position in the first $n-2$ 1-bits of x_1 and x_2 (tie and half chance to win).
  (3) The noise flips the last 0-bits of x_1, and does not flip the 0-bit of x_2.
  (4) The noise flips both the 0-bits of x_1 and x_2 (tie and half chance to win).
  (5) the noise flips the $(n-1)$-th 1-bit of x_1, and flips the last 0-bits of x_2 (tie and half chance to win).

  Thus,

$$\Pr(E_{11})$$
$$\geq q^2\left(\sum_{i=2}^{n-1}\left(\frac{i-1}{n}\cdot\frac{1}{n}\right)+\frac{n-2}{2n^2}+\frac{1}{n}\left(1-\frac{1}{n}\right)+\frac{1}{2n^2}+\frac{1}{2n^2}\right)=\frac{q^2}{2}.$$

  Therefore, we can get $\Pr(E_{11}) \geq q^2/2$ for all $j \leq n-2$.

By combining all four cases above and $j \leq n-2$, we can get

$$\Pr(E) \geq \Pr(E_{00}) + \Pr(E_{01}) + \Pr(E_{10}) + \Pr(E_{11})$$
$$\geq 1-\frac{j+2}{n}q+\left(\frac{j+2}{n}-1\right)q^2+\frac{q^2}{2}$$
$$\geq 1-\frac{j+2}{n}(1-q)q-q^2+\frac{q^2}{2} \geq 1-(1-q)q-q^2+\frac{q^2}{2}$$
$$=1-q+\frac{q^2}{2}=\frac{1}{2}+\frac{1}{2}-q(1-\frac{q}{2})=\frac{1}{2}+\theta.$$

(c) To estimate a lower bound for $\Pr(E)$ on the OneMax problem in the bit-wise noise model (p), we pessimistically assume that $x_1 \in A_{j+1}$ and $x_2 \in A_j$, such that $f(x_1) = f(x_2) + 1$, otherwise x_1 will always win. There exists at least one bit-position i, such that x_1 has a 1-bit in position i and x_2 has a 0-bit in position i. The remaining bits of x_1 and x_2 have the same number of 1-bits. Therefore, the bits after noise of x_1 and x_2 in position i decide the comparison result. Let x_1' and x_2' be two substrings obtained by removing position i from x_1 and x_2 respectively. Since each bit is flipped independently, we know that $f^n(x_1'), f^n(x_2') \sim \mathrm{Bin}\,(n-1-j, p) + \mathrm{Bin}\,(j, (1-p))$ which are Poisson-binomially distributed random variables with variance $\sigma^2 = (1-p)p(n-1)$. Then we apply Lemma 16 with $\sigma = \sqrt{(1-p)p(n-1)} \leq \sqrt{pn}$ and $d = 2$ to obtain a lower bound for

$$\Pr(f^n(x_1') = f^n(x_2')) \geq \frac{(1-1/2^2)^2}{2\cdot 2\sqrt{2pn}+1} \geq \frac{9}{64\sqrt{2pn}+16}. \qquad (9)$$

By symmetry, we know that $\Pr(f^n(x_1') > f^n(x_2')) = \Pr(f^n(x_1') < f^n(x_2'))$. Let $a = \Pr(f^n(x_1') = f^n(x_2'))$ and $b = Pr(f^n(x_1') > f^n(x_2'))$, then we can get $a = 1 - 2b$. Thus,

$$\Pr(E) = b + a\left((1-p)^2 + 2\cdot\frac{1}{2}\cdot p(1-p)\right)$$
$$=\frac{1}{2}(1-a)+a\cdot(1-p)=\frac{1}{2}+\left(\frac{1}{2}-p\right)a$$

by Eq. (9),

$$\geq \frac{1}{2}+\frac{9\,(1/2-p)}{64\sqrt{2pn}+16}=\frac{1}{2}+\theta.$$

(d) To estimate a lower bound for $\Pr(E)$ on the LeadingOnes problem in the bit-wise noise model (p), we pessimistically assume that $x_1 \in A_{j+1}$ and $x_2 \in A_j$. We also pessimistically assume that the suffix of x_1, i.e. the bits after the $(j+2)$-th position, is all 0-bits, and the suffix of x_2, i.e. the bits after the $(j+1)$-th position, is all 1-bits, which is the worst case since the noise flipping $(j+1)$-th bit of x_2 to achieve the optimum which leads to incorrect comparison while the noise only can at most increase 1 fitness for x_1. We distinguish 3 cases to estimate $\Pr(E)$:

- Let E_0 be the event that $f^n(x_1) \geq j+1$ and $f^n(x_2) \leq j$, then $\Pr(E_0) = (1-p)^{j+1}(1-p) + (1-p)^{j+1}p\left(1-(1-p)^j\right) = \left(1-p(1-p)^j\right)(1-p)^{j+1}$.
- Let E_{1i} be the event that $f^n(x_1) = i$ and $f^n(x_2) \leq i-1$ for any $i \in [1, j]$, then $\Pr(E_1) = \sum_{i=0}^{j}\Pr(E_{1i}) = \sum_{i=0}^{j}(p(1-p)^i \cdot \left(1-(1-p)^i\right)\,) = p\left(\sum_{i=0}^{j}(1-p)^i - \sum_{i=0}^{j}(1-p)^{2i}\right)$, by

the sum of a geometric series,

$$\begin{aligned}\Pr(E_1) &= p\left(\frac{1-(1-p)^{j+1}}{1-(1-p)} - \frac{1-(1-p)^{2(j+1)}}{1-(1-p)^2}\right)\\ &= \frac{1}{2} - \frac{p}{2(2-p)} - (1-p)^{j+1} + \frac{1}{2-p}(1-p)^{2(j+1)}.\end{aligned}$$

- Let E_{2i} be the event that $f^n(x_1) = i$ and $f^n(x_2) = i$ for any $i \in [0, j]$ then x_1 is selected uniformly, then $\Pr(E_2) = \frac{1}{2}\sum_{i=0}^{j} \Pr(E_{2i}) = \frac{1}{2}\sum_{i=0}^{j} p^2(1-p)^{2i}$, by the sum of a geometric series, $\Pr(E_2) = \frac{1}{2}p^2\left(\frac{1-(1-p)^{2(j+1)}}{1-(1-p)^2}\right) = \frac{p}{2(2-p)} - \frac{p}{2(2-p)}(1-p)^{2(j+1)}$.

By combining all three cases above, we can get

$$\begin{aligned}&\Pr(E) \geq \Pr(E_0) + \Pr(E_1) + \Pr(E_2)\\ &= \left(1-p(1-p)^j\right)(1-p)^{j+1} + \frac{1}{2} - \frac{p}{2(2-p)} - (1-p)^{j+1}\\ &\quad + \frac{1}{2-p}(1-p)^{2(j+1)} + \frac{p}{2(2-p)} - \frac{p}{2(2-p)}(1-p)^{2(j+1)}\\ &= \frac{1}{2} + \frac{(1-p)^{2(j+1)}}{2-p} - \frac{p}{2(2-p)}(1-p)^{2(j+1)} - p(1-p)^{2j+1}\\ &= \frac{1}{2} + \frac{1}{2}(1-p)^{2j+2} - p(1-p)^{2j+1} = \frac{1}{2} + \left(\frac{1}{2} - \frac{3}{2}p\right)(1-p)^{2j+1}\end{aligned}$$

by $\left((1-x)^{1/x-1}\right)^y \geq e^{-y}$ (Lemma 12),

$$\geq \frac{1}{2} + \left(\frac{1}{2} - \frac{3}{2}p\right)e^{-(2j+1)\frac{p}{1-p}} > \frac{1}{2} + \left(\frac{1}{2} - \frac{3}{2}p\right)e^{-2(j+2)\frac{p}{1-p}}$$

by $0 \leq j \leq n-2$ and $p \in [0, 1/3)$,

$$\geq \frac{1}{2} + \left(\frac{1}{2} - \frac{3}{2}p\right)e^{-3np} \geq \frac{1}{2} + \theta.$$

(e) To estimate a lower bound for $\Pr(E)$ on the OneMax and LeadingOnes problem in the Gaussian noise model (σ^2), we pessimistically assume that $x_1 \in A_{j+1}$ and $x_2 \in A_j$. Let $X \sim \mathcal{N}(0, 2\sigma^2)$ be a random variable, then

$$\Pr(E) \geq \Pr(f^n(x_1) - f^n(x_2) > 0) = \Pr(X > -1) = \Pr(X < 1)$$

by Lemma 14 with $x = 1$ and standard deviation is $\sqrt{2}\sigma$,

$$\begin{aligned}&> 1 - 1/\sqrt{\pi/(\sqrt{2}\sqrt{2}\sigma) + 4}\\ &= \left(1 - \left(1/\sqrt{\pi/(2\sigma) + 4}\right)^2\right)/\left(1 + 1/\sqrt{\pi/(2\sigma) + 4}\right)\\ &> \left(\frac{\pi/(2\sigma) + 3}{\pi/(2\sigma) + 4}\right)/\frac{3}{2} = \frac{1 + 6\sigma/\pi}{1 + 8\sigma/\pi} \cdot \frac{4}{3} \cdot \frac{1}{2} = \frac{1}{2} + \frac{1}{6 + 48\sigma/\pi}\\ &= \frac{1}{2} + \theta.\end{aligned}$$

□

B.2 Proof of Theorem 4

Proof. We apply Theorem 2 to prove Theorem 4. We first partition the search space into levels. We use the partition $A_j := \{x \in \{0,1\}^n \mid f(x) = j\}$ for all $j \in [0..n]$. By constants $q \in [0, 1]$ and $n_0 \in [3, \infty)$, we can get $1/12 < \theta \leq 1/2$ which satisfies the assumption in Theorem 2.

By case (a) of Lemma 3, we get $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) > 1/2 + \theta$, then the condition (C2) of Theorem 2 holds.

To verify condition (C1), we need to estimate the probability of sampling individuals beyond the current level of the population. We assume that there is an individual $z \in A_j$ where $j \in [0..n-1]$, and let y be obtained from z by the mutation operator with mutation rate χ/n. We only consider the case that all 1-bits is not flipped and one of 0-bits is flipped after mutation, and then by Lemma 13 following

$$\begin{aligned}&\Pr(y \in A_{\geq j+1} \mid z \in A_j) > \left(1 - \frac{\chi}{n}\right)^j \frac{\chi}{n}(n-j)\\ &\geq e^{-\chi}(1 - o(1))(n-j)\chi/n =: h_j \in \Omega((n-j)\chi/n).\end{aligned}$$

Then we compute the population size required by condition (C3). Let $\xi \in (0, 1/16)$ be a constant, then

$$\lambda > \frac{4(1 + o(1))}{\theta^2\xi(1-\zeta)^4}\ln\left(\frac{128(m+1)}{\theta^2\xi(1-\zeta)^4\min\{h_j\}}\right) = O(\log(n/\chi)).$$

Condition (C3) is satisfied by $\lambda \geq c\log(n/\chi)$ for a sufficiently large constant c.

Finally, all conditions of Theorem 2 hold and the expected time to reach the optimum is no more than

$$\begin{aligned}&E[T]\\ &\leq \frac{16(1 + o(1))}{\theta^2\xi(1-\zeta)^2}\left(\lambda\sum_{j=0}^{m-1}\ln\left(\frac{6}{\xi(1-\zeta)^2 h_j}\right) + \frac{1}{\xi(1-\zeta)^2}\sum_{j=0}^{m-1}\frac{1}{h_j}\right)\\ &= O\left(\lambda\sum_{j=0}^{m-1}\ln\left(\frac{n}{(n-j)\chi}\right) + \sum_{j=0}^{m-1}\frac{n}{(n-j)\chi}\right)\\ &= O\left(\lambda\ln\left(\prod_{j=0}^{m-1}\frac{n}{(n-j)\chi}\right) + n\sum_{j=0}^{m-1}\frac{1}{(n-j)\chi}\right)\\ &= O\left(\lambda\ln\left(\frac{n^n}{n!\,\chi^n}\right) + n\log(n)/\chi\right)\end{aligned}$$

using the lower bound $n! > (n/e)^n$,

$$= O(\lambda n\log(1/\chi) + n\log(n)/\chi).$$

□

B.3 Proof of Theorem 5

Proof. We apply Theorem 2 to prove Theorem 5. We first partition the search space into levels. We use the partition $A_j := \{x \in \{0,1\}^n \mid f(x) = j\}$ for all $j \in [0..n]$. By $q \in [0, 1)$, we know that $0 < \theta \leq \frac{1}{2}$ which satisfies the assumption in Theorem 2.

By case (b) of Lemma 3, we get $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) > 1/2 + \theta$, then the condition (C2) of Theorem 2 holds.

To verify condition (C1), we need to estimate the probability of sampling individuals beyond the current level of the population. We assume that there is an individual $z \in A_j$ where $j \in [0..n-1]$, and let y be obtained from z by the mutation operator with mutation rate χ/n. We only consider the case that the first 0-bit is flipped and other bits are not flipped, and then by Lemma 13 following,

$$\Pr(y \in A_{\geq j+1} \mid z \in A_j) \geq \left(1 - \frac{\chi}{n}\right)^{n-1} \frac{\chi}{n} \geq e^{-\chi}\left(1 - o(1)\right) \frac{\chi}{n} \quad (10)$$
$$=: h_j = \Omega\left(\chi/n\right).$$

Then we compute the population size required by condition (C3). Let $\xi \in (0, 1/16)$ be a constant, then

$$\lambda > \frac{4\left(1 + o(1)\right)}{\theta^2 \xi (1 - \zeta)^4} \ln\left(\frac{128(m+1)}{\theta^2 \xi (1 - \zeta)^4 \min\{h_j\}}\right) = O\left(\log\left(n/\chi\right)\right)$$

Condition (C3) is satisfied by $\lambda \geq c \log\left(n/\chi\right)$ for a sufficiently large constant c.

Finally, all conditions of Theorem 2 hold and the expected time to reach the optimum is no more than

$$E[T] < \frac{16\left(1 + o(1)\right)}{\theta^2 \xi (1 - \zeta)^2} \left(\lambda \sum_{j=0}^{m-1} \ln\left(\frac{6}{\xi (1 - \zeta)^2 h_j}\right) + \frac{1}{\xi (1 - \zeta)^2} \sum_{j=0}^{m-1} \frac{1}{h_j}\right)$$
$$= O\left(\lambda \sum_{j=0}^{m-1} \ln\left(n/\chi\right) + \sum_{j=0}^{m-1} n/\chi\right) = O\left(n\lambda \log\left(n/\chi\right) + n^2/\chi\right).$$

□

B.4 Proof of Theorem 6

Proof. We apply Theorem 2 to prove Theorem 6. We first partition the search space into levels. We use the partition $A_j := \{x \in \{0,1\}^n \mid f(x){=}j\}$ for all $j \in [0..n]$. Since $p \in (0, 1/2)$, we know that $0 < \theta < 9/32$ which satisfies the assumption in Theorem 2.

By case (c) of Lemma 3, we get $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) > 1/2 + \theta$, then the condition (C2) of Theorem 2 holds.

To verify condition (C1), we need to estimate the probability of sampling individuals beyond the current level of the population. We assume that there is an individual $z \in A_j$ where $j \in [0..n-1]$, and let y be obtained from z by the mutation operator with mutation rate χ/n. We only consider the case that no 1-bits are flipped and one of the 0-bits is flipped after mutation:

$$\Pr(y \in A_{\geq j+1} \mid z \in A_j) > \left(1 - \frac{\chi}{n}\right)^j \frac{\chi}{n}(n - j)$$
$$\geq \left(1 - \frac{\chi}{n}\right)^n \frac{\chi}{n}(n - j) \geq e^{-\chi}\left(1 - \frac{\chi^2}{n}\right)\frac{\chi}{n}(n - j)$$
$$\geq e^{-\chi}\left(1 - o(1)\right)(n - j)\chi/n =: h_j = \Omega\left((n - j)\chi/n\right).$$

Then we compute the population size required by condition (C3). Let $\xi \in (0, 1/16)$ be a constant, then

$$\lambda > \frac{4\left(1 + o(1)\right)}{\theta^2 \xi (1 - \zeta)^4} \ln\left(\frac{128(m+1)}{\theta^2 \xi (1 - \zeta)^4 \min\{h_j\}}\right) = O\left(\frac{1}{\theta^2}\ln\left(\frac{n^2}{\chi\theta^2}\right)\right)$$
$$= O\left(\frac{\log\left(n/(\chi\theta)\right)}{\theta^2}\right) = O\left(\frac{1 + pn}{(1 - 2p)^2}\log\left(\frac{n}{(1 - 2p)\chi}\right)\right).$$

Condition (C3) is satisfied by $\lambda \geq c\frac{1+pn}{(1-2p)^2}\log\left(\frac{n}{(1-2p)\chi}\right)$ for a sufficiently large constant c.

Finally, all conditions of Theorem 2 hold and the expected time to reach the optimum is no more than

$$E[T] \leq \frac{16\left(1 + o(1)\right)}{\theta^2 \xi (1 - \zeta)^2} \left(\lambda \sum_{j=0}^{m-1} \ln\left(\frac{6}{\xi (1 - \zeta)^2 h_j}\right) + \frac{1}{\xi (1 - \zeta)^2} \sum_{j=0}^{m-1} \frac{1}{h_j}\right)$$
$$= O\left(\frac{1}{\theta^2}\left(\lambda \sum_{j=0}^{m-1} \ln\left(\frac{n}{(n - j)\chi}\right) + \sum_{j=0}^{m-1} \frac{n}{(n - j)\chi}\right)\right)$$
$$= O\left(\frac{1}{\theta^2}\left(\lambda \ln\left(\frac{n^n}{\chi^n \cdot n!}\right) + \frac{n}{\chi}\log(n)\right)\right)$$

using the lower bound $n! > (n/e)^n$,

$$= O\left(\frac{n\lambda}{\theta^2}\log\left(\frac{1}{\chi}\right) + \frac{n}{\chi\theta^2}\log(n)\right)$$
$$= O\left(\frac{n(1 + pn)}{(1 - 2p)^2}\left(\lambda \log\left(\frac{1}{\chi}\right) + \frac{\log(n)}{\chi}\right)\right).$$

□

B.5 Proof of Theorem 7

Proof. We apply Theorem 2 to prove Theorem 7. We first partition the search space into levels. We use the partition $A_j :=\{ x \in \{0,1\}^n \mid f(x) = j\}$ for all $j \in [0..n]$. Since $p \in [0, 1/3)$, we know that $0 < \theta \leq 1/2$ satisfies the assumption in Theorem 2.

By case (d) of Lemma 3, we get $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) > 1/2 + \theta$, then the condition (C2) of Theorem 2 holds.

To verify condition (C1), we need to estimate the probability of sampling individuals beyond the current level of the population. We assume that there is an individual $z \in A_j$ where $j \in [0..n-1]$, and let y be obtained from z by the mutation operator with mutation rate χ/n. We only consider the case that the first 0-bit is flipped and other bits are not flipped, and then by Lemma 13 following,

$$\Pr(y \in A_{\geq j+1} \mid z \in A_j) \geq \left(1 - \frac{\chi}{n}\right)^{n-1} \frac{\chi}{n} \geq e^{-\chi}\left(1 - o(1)\right)\frac{\chi}{n} =: h_j$$
$$= \Omega\left(\chi/n\right).$$

Then we compute the population size required by condition (C3). Let $\xi \in (0, 1/16)$ be a constant, then

$$\lambda > \frac{4\left(1 + o(1)\right)}{\theta^2 \xi (1 - \zeta)^4} \ln\left(\frac{128(m+1)}{\theta^2 \xi (1 - \zeta)^4 \min\{h_j\}}\right)$$
$$= O\left(\log\left(n/\chi\right)/\theta^2\right) = O\left(\frac{e^{6np}}{(1 - 3p)^2}\log\left(\frac{n}{\chi}\right)\right).$$

Condition (C3) is satisfied by $\lambda \geq c\frac{e^{6np}}{(1-3p)^2}\log\left(\frac{n}{\chi}\right)$ for a sufficiently large constant c.

Finally, all conditions of Theorem 2 hold and the expected time to reach the optimum is no more than

$$E[T] < \frac{16\left(1 + o(1)\right)}{\theta^2 \xi (1 - \zeta)^2} \left(\lambda \sum_{j=0}^{m-1} \ln\left(\frac{6}{\xi (1 - \zeta)^2 h_j}\right) + \frac{1}{\xi (1 - \zeta)^2} \sum_{j=0}^{m-1} \frac{1}{h_j}\right)$$
$$= O\left(\frac{1}{\theta^2}\left(\lambda \sum_{j=0}^{m-1} \ln\left(\frac{n}{\chi}\right) + \sum_{j=0}^{m-1} \frac{n}{\chi}\right)\right)$$
$$= O\left(\frac{1}{\theta^2}\left(n\lambda \log\left(\frac{n}{\chi}\right) + \frac{n^2}{\chi}\right)\right) = O\left(\frac{n\lambda\log\left(n/\chi\right)}{\theta^2} + \frac{n^2}{\chi\theta^2}\right)$$
$$= O\left(\frac{ne^{6np}}{(1 - 3p)^2}\left(\lambda \log\left(\frac{n}{\chi}\right) + \frac{n}{\chi}\right)\right).$$

□

B.6 Proof of Theorem 8

Proof. We apply Theorem 2 to prove Theorem 8. We first partition the search space into levels. We use the partition $A_j := \{ x \in \{0,1\}^n \mid f(x) = j \}$ for all $j \in [0..n]$. By $\sigma \in \text{poly}(n)$, we can get $0 < \theta < \frac{1}{6}$ which satisfies the assumption in Theorem 2.

By case (e) of Lemma 3, we get $\Pr(f^n(x_1) > f^n(x_2)) + \frac{1}{2}\Pr(f^n(x_1) = f^n(x_2)) > 1/2 + \theta$, then the condition (C2) of Theorem 2 holds.

To verify condition (C1), we need to estimate the probability of sampling individuals beyond the current level of the population. We assume that there is an individual $z \in A_j$ where $j \in [0..n-1]$, and let y be obtained from z by the mutation operator with mutation rate χ/n. We only consider the case that all 1-bits is not flipped and one of 0-bits is flipped after mutation for OneMax, and then by Lemma 13 following,

$$\begin{aligned}\Pr(y \in A_{\geq j+1} \mid z \in A_j) &> \left(1 - \frac{\chi}{n}\right)^j \frac{\chi}{n}(n-j) \\ &\geq e^{-\chi}\,(1 - o(1))\,(n-j)\chi/n =: h_j\end{aligned}$$

For LeadingOnes, we only consider the case that the first 0-bit is flipped and other bits are not flipped, and then by Lemma 13 following,

$$\Pr(y \in A_{\geq j+1} \mid z \in A_j) \geq \left(1 - \frac{\chi}{n}\right)^{n-1} \frac{\chi}{n} \geq e^{-\chi}\,(1 - o(1))\,\frac{\chi}{n} =: h_j.$$

Then, we get $h_j \in \Omega\left((n-j)\chi/n\right)$ and $h_j \in \Omega\left(\chi/n\right)$ for OneMax and LeadingOnes respectively.

Then we compute the population size required by condition (C3). Let $\xi \in (0, 1/16)$ be a constant and we know that $\min\{h_j\} \in \Omega\left(\chi/n\right)$ for both problems, then

$$\lambda > \frac{4\,(1+o(1))}{\theta^2\xi(1-\zeta)^4}\ln\left(\frac{128(m+1)}{\theta^2\xi(1-\zeta)^4\min\{h_j\}}\right) = O\left(\sigma^2\log(n/\chi)\right)$$

Condition (C3) is satisfied by $\lambda \geq c\sigma^2\log(n/\chi)$ for a sufficiently large constant c.

Finally, all conditions of Theorem 2 hold and the expected time on OneMax to reach the optimum is no more than

$$\begin{aligned}E[T] &\leq \frac{16\,(1+o(1))}{\theta^2\xi(1-\zeta)^2}\left(\lambda\sum_{j=0}^{m-1}\ln\left(\frac{6}{\xi(1-\zeta)^2h_j}\right) + \frac{1}{\xi(1-\zeta)^2}\sum_{j=0}^{m-1}\frac{1}{h_j}\right) \\ &= O\left(\sigma^2\left(\lambda\sum_{j=0}^{m-1}\ln\left(\frac{n}{(n-j)\chi}\right) + \sum_{j=0}^{m-1}\frac{n}{(n-j)\chi}\right)\right) \\ &= O\left(\sigma^2\left(\lambda\ln\left(\frac{n^n}{\chi^n n!}\right) + n\log(n)/\chi\right)\right)\end{aligned}$$

using the lower bound $n! > (n/e)^n$,

$$= O\left(\sigma^2\lambda n\log(1/\chi) + \sigma^2 n\log(n)/\chi\right),$$

and on LeadingOnes,

$$\begin{aligned}E[T] &= O\left(\sigma^2\left(\lambda\sum_{j=0}^{m-1}\ln(\sigma n) + \sum_{j=0}^{m-1}\sigma n\right)\right) \\ &= O\left(\sigma^2\lambda n\log(n/\chi) + \sigma^2 n^2/\chi\right).\end{aligned}$$

□