Analysis of the (1+1) EA on LeadingOnes with Constraints

Tobias Friedrich
friedrich@hpi.de
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany

Timo Kötzing
timo.koetzing@hpi.de
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany

Aneta Neumann
aneta.neumann@adelaide.edu.au
Optimisation and Logistics
The University of Adelaide
Adelaide, Australia

Frank Neumann
frank.neumann@adelaide.edu.au
Optimisation and Logistics
The University of Adelaide
Adelaide, Australia

Aishwarya Radhakrishnan
aishwarya.radhakrishnan@hpi.de
Hasso Plattner Institute
University of Potsdam
Potsdam, Germany

ABSTRACT

Understanding how evolutionary algorithms perform on constrained problems has gained increasing attention in recent years. In this paper, we study how evolutionary algorithms optimize constrained versions of the classical LeadingOnes problem. We first provide a run time analysis for the classical (1+1) EA on the LeadingOnes problem with a deterministic cardinality constraint, giving $\Theta(n(n-B)\log(B)+n^2)$ as the tight bound. Our results show that the behaviour of the algorithm is highly dependent on the constraint bound of the uniform constraint. Afterwards, we consider the problem in the context of stochastic constraints and provide insights using experimental studies on how the (μ+1) EA is able to deal with these constraints in a sampling-based setting.

KEYWORDS

Evolutionary algorithms, chance constraint optimization, run time analysis, theory.

1 INTRODUCTION

Evolutionary algorithms [6] have been used to tackle a wide range of combinatorial and complex engineering problems. Understanding evolutionary algorithms from a theoretical perspective is crucial to explain their success and give guidelines for their application. The area of run time analysis has been a major contributor to the theoretical understanding of evolutionary algorithms over the last 25 years [4, 11, 19]. Classical benchmark problems such as OneMax and LeadingOnes have been analyzed in a very detailed way, showing deep insights into the working behaviour of evolutionary algorithms for these problems. In real-world settings, problems that are optimized usually come with a set of constraints which often limits the resources available. Studying classical benchmark problems even with an additional simple constraint such as a uniform constraint, which limits the number of elements that can be chosen in a given benchmark function, poses significant new technical challenges for providing run time bounds of even simple evolutionary algorithms such as the (1+1) EA.

OneMax and the broader class of linear functions [5] have played a key role in developing the area of run time analysis during the last 25 years, and run time bounds for linear functions with a uniform constraint have been obtained [7, 17]. It has been shown in [7] that the (1+1) EA needs exponential time optimize OneMax under a specific linear constraint which points to the additional difficulty which such constraints impose on the search process. Tackling constraints by taking them as additional objectives has been shown to be quite successful for a wide range of problems. For example, the behaviour of evolutionary multi-objective algorithms has been analyzed for submodular optimization problems with various types of constraints [20, 21]. Furthermore, the performance of evolutionary algorithms for problems with dynamic constraints has been investigated in [22, 23].

Another important area involving constraints is chance constrained optimization, which deals with stochastic components in the constraints. Here, the presence of stochastic components in the constraints makes it challenging to guarantee that the constraints are not violated at all. Chance-constrained optimization problems [2, 13] are an important class of the stochastic optimization problems [1] that optimize a given problem under the condition that a constraint is only violated with a small probability. Such problems occur in a wide range of areas, including finance, logistics and engineering [9, 12, 14, 29]. Recent studies of evolutionary algorithms for chance-constrained problems focused on a classic knapsack problem where the uncertainty lies in the probabilistic constraints [26, 27]. Here, the aim is to maximise the deterministic profit subject to a constraint which involves stochastic weights and where the knapsack capacity bound can only be violated with a small probability of at most α. A different stochastic version of the knapsack problem has been studied in [16]. Here profits involve uncertainties and weights are deterministic. In that work, Chebyshev and Hoeffding-based fitness functions have been introduced and evaluated. These fitness functions discount expected profit values based on uncertainties of the given solutions.

Theoretical investigations for problems with chance constraints have gained recent attention in the area of run time analysis. This includes studies for montone submodular problems [15] and special instances of makespan scheduling [24]. Furthermore, detailed run time analyses have been carried out for specific classes of instances for the chance constrained knapsack problem [18, 28].

1.1 Our contribution

In this paper, we investigate the behaviour of the (1+1) EA for the classical LeadingOnes problem with additional constraints. We first study the behaviour for the case of a uniform constraint which limits the number of 1-bits that can be contained in any feasible solution. Let B be the upper bound on the number of 1-bits that any feasible solution can have. Then the optimal solutions consists of exactly B leading 1s and afterwards only 0s. The search for the (1+1) EA is complicated by the fact that when the current solution consists of $k < B$ leading 1s, additional 1-bits not contributing to the fitness score at positions $k+2, \ldots, n$ might make solutions infeasible. We provide a detailed analysis of such scenarios in dependence of the given bound B.

Specifically, we show a tight bound of $\Theta(n^2+n(n-B)\log(B))$ (see Corollary 6). Note that [7] shows the weaker bound of $O(n^2 \log(B))$, which, crucially, does not give insight into the actual optimization process at the constraint. Our analysis shows in some detail how the search progresses. In the following discussion, for the current search point of the algorithm, we call the part of the leading 1s the *head* of the bit string, the first 0 the *critical bit* and the remaining bits the *tail*. While the size of the head is less than $B - (n - B)$, optimization proceeds much like for unconstrained LeadingOnes; this is because the bits in the tail of size about $2(n - B)$ are (almost) uniformly distributed, contributing roughly a number of $n-B$ many 1s additionally to the $B - (n - B)$ many 1s in the head. This stays in sum (mostly) below the cardinality bound B, occasional violations changing the uniform distribution of the tail to one where bits in the tail are 1 with probability a little less than $1/2$ (see Lemma 3).

Once the threshold of $B - (n - B)$ many 1s in the head is passed, the algorithm frequently runs into the constraint. For a phase of equal LeadingOnes value, we consider the random walk of the number of 1s of the bit string of the algorithm. This walk has a bias towards the bound B (its maximal value), where the bias is light for LeadingOnes-values just a bit above $B - (n - B)$ and getting stronger as this value approaches B. Since progress is easy when not at the bound of B many 1s in the bit string (by flipping the critical bit and no other) and difficult otherwise (additionally to flipping the critical bit, a 1 in the tail needs to flip), the exact proportion of time that the walk spends in states of less than B versus exactly B many 1s is very important. In the final proofs, we estimate these factors and have corresponding potential functions reflecting gains (1) from changing into states of less than B many 1s and (2) gaining a leading 1. Bounding these gains appropriately lets us find asymptotically matching upper and lower bounds using the additive drift theorem [10].

In passing we note that two different modifications of the setting yield a better time of $O(n^2)$. First, this time is sufficient to achieve a LeadingOnes-values of $B - c(n - B)$ for any $c > 0$ (see Corollary 7). Second, considering the number of 1s as a secondary objective (to be minimized) gives an optimization time of $O(n^2)$ (see Theorem 8).

Afterwards, we turn to stochastic constraints and investigate an experimental setting that is motivated by recent studies in the area of chance constraints. We consider LeadingOnes with a stochastic knapsack chance constraint, where the weights of a linear constraint are chosen from a given distribution. In the first setting, the weight of each item is chosen independently according to a Normal distribution $N(\mu, \sigma^2)$. A random sample of weights is feasible if the sum of the chosen sampled weights does not exceed a given knapsack bound B. In any iteration, all weights are resampled independently for all evaluated individuals. Our goal is to understand the maximal stable LeadingOnes value that the algorithm obtains. In the second setting which we study empirically, the weights are deterministically set to 1 and the bound is chosen uniformly at random within an interval $[B - \epsilon, B + \epsilon]$, where $\epsilon > 0$ specifies the uncertainty around the constraint bound. For both settings, we examine the performance of the $(1 + 1)$ EA and $(10 + 1)$-EA for different values of B and show that a larger parent population has a highly positive effect for these stochastic settings.

The paper is structured as follows. In Section 2, we introduce the problems and algorithms that we study in this paper. We present our run time analysis for the LeadingOnes problem with a deterministic uniform constraint in Section 3. In section 4, we discuss a way to obtain $\Theta(n^2)$ bound on the run time for the same problem and report on our empirically investigations for the stochastic settings in Section 5. Finally, we finish with some concluding remarks. Note that some proofs are ommitted due to space constraints.

2 PRELIMINARIES

In this section we define the objective function, constraints and the algorithms used in our analysis. With $|x|_1$ we denote the number of 1s in a bit string $x \in \{0, 1\}^n$.

2.1 Cardinality Constraint

Let $f : \{0, 1\}^n \to \mathbb{R}$, $B \leq n$ and for $x \in \{0, 1\}^n$, let x_i denote the i -th bit of x. In this paper, optimizing f with cardinality constraint B means finding, $\max_{x \in \{0,1\}^n} f(x)$ s.t $\sum_{i=1}^{n} x_i \leq B$.

2.2 Stochastic Constraint

Let $f : \{0, 1\}^n \to \mathbb{R}$, $B \leq n$ and for $x \in \{0, 1\}^n$, let x_i denote the i -th bit of x. In this paper we empirically analyse the following normal stochastic constraint with uncertainty in the weights optimization problem,

$$\max_{x \in \{0,1\}^n} f(x) \text{ s.t } \sum_{i=1}^{n} w_i \cdot x_i \leq B, \text{ where } w_i \sim N(\mu, \sigma^2).$$

Let $f : \{0, 1\}^n \to \mathbb{R}$, $B \leq n$ and for $x \in \{0, 1\}^n$, let x_i denote the i -th bit of x. In this paper we also empirically analyse the following uniform stochastic constraint with uncertainty in the bound optimization problem,

$$\max_{x \in \{0,1\}^n} f(x) \text{ s.t } |x|_1 \leq y, \text{ where } y \sim U(B - \epsilon, B + \epsilon).$$

2.3 Objective Function

We consider the LeadingOnes function as our objective with cardinality and stochastic constraints for our analysis.

LeadingOnes : $\{0, 1\}^n \to \mathbb{R}$, is a function which maps a bit string of length n to number of 1s before the first 0 in the bit string. For every $x \in \{0, 1\}^n$, $\text{LeadingOnes}(x) = \sum_{i=1}^{n} \prod_{j=1}^{i} x_j$.

2.4 (μ+1) EA

The (μ+1) EA on a real valued fitness function f with constraint B is given in Algorithm 1. The (μ+1) EA at each iteration maintains a population of size μ. The initial population P_0 has μ random bit strings chosen uniformly. At each iteration $t > 0$, a bit string is chosen uniformly at random from P_t followed by a mutation operation which flips each bit of the chosen bit string with probability $\frac{1}{n}$. The mutated bit string is added to P_t and the bit string with the least fitness among the $\mu + 1$ individuals is removed. Since we can also sample a bit string which violates the constraint, we consider the following function for optimization.

$$g(x) = \begin{cases} f(x), & \text{if } |x|_1 < B; \\ B - |x|, & \text{otherwise.} \end{cases}$$

Algorithm 1: (μ+1) EA on fitness function f with constraint B

```
$P_0 \leftarrow \mu$ individuals from $\{0,1\}^n$ chosen u.a.r.;
$t = 0$;
while stopping criterion not met do
    $x \leftarrow$ uniform random bit string from $P_t$;
    $y \leftarrow$ flip each bit of $x$ independently with probab. $1/n$;
    $P_t = P_t \cup \{y\}$;
    $P_{t+1} = P_t \setminus \{$an individual $x \in P_t$ with least $g(x)$ value$\}$;
    $t = t + 1$;
```

3 UNMODIFIED SETTING

In this section we give a tight analysis of the (1+1) EA on the objective LeadingOnes with cardinality constraint B.

We start with a technical lemma which we need for our proof of the upper bound.

Lemma 1. *For $t \geq 0$, let x_t denote the parent bit string at t-th iteration while (1+1) EA is optimizing LeadingOnes with the cardinality constraint B. And for $t > 0$, let A_t denote the event that $|x_{t+1}|_1 = B$ and $LO(x_{t+1}) = LO(x_t)$. Then $Pr(A_t \mid |x_t|_1 < B) \leq \frac{n-B}{n}$.*

Proof. First note that, if $|x_t|_1 = k < B$ and C_t denote the event that x_{t+1} is formed by flipping $B - k$ number of 0 bits to 1 out of $n - k - 1$ (except the left most 0) number of 0 bits, then

$$Pr(A_t \mid |x_t|_1 < B) \leq Pr(C_t \mid |x_t|_1 < B).$$

The event A_t is a sub-event of C_t, since in the event C_t we do not have any restriction on the bits other than $B - k$ number of 0 bits out of $n - k - 1$ number of them and we have to flip at least $B - k$ number of 0 bits to 1 to get the desired x_{t+1} in the event A_t. Hence,

$$\begin{aligned} Pr(C_t) &= \binom{n-k-1}{B-k}\left(\frac{1}{n}\right)^{B-k} \\ &= \frac{(n-k-1)\cdot(n-k-2)\cdots(n-B)}{1\cdot 2\cdots(B-k)}\cdot\left(\frac{1}{n}\right)^{B-k} \leq \frac{n-B}{n}. \end{aligned}$$

The last inequality holds because, for every $r > 0$, $\frac{n-k-r}{n} \leq 1$. □

In the Theorem 2 below we give an upper bound on the expected run time of the (1+1) EA on LeadingOnes with cardinality constraint B. Later we show that this bound is tight by proving a matching lower bound.

Theorem 2. *Let $n, B \in \mathbb{N}$ and $B < n$. Then the expected optimization time of the (1+1) EA on LeadingOnes with cardinality constraint B is $O\left(n^2 + n(n-B)\log B\right)$.*

Proof. From [8, Lemma 3], we know that the (1+1) EA is expected to find a feasible solution within $O(n\log(n/B))$ iterations. Now we calculate how long it takes in expected value to find the optimum after a feasible solution is sampled.

To do this, we construct a potential function that yields an drift value greater than 1 at each time t until the optimum is found. For $i \in \{0, \cdots, B\}$, let $g_B(i)$ be the potential of a bit string $x \in \{0,1\}^n$ with exactly B number of 1s and $LO(x) = \text{LeadingOnes}(x) = i$. For $i \in \{0, \cdots, B-1\}$, let $g_{<B}(i)$ be the potential of a bit string $x \in \{0,1\}^n$ with less than B number of 1s and $LO(x) = i$.

Let $g_B(0) = 0$ and $g_{<B}(0) = \frac{en}{B}$. And for every $i \in \{1, \cdots, B\}$, let

$$g_B(i) = en\left(1 + \frac{e\cdot(n-B)}{B-i+1}\right) + g_{<B}(i-1),$$

and for every $i \in \{1, \cdots, B-1\}$, let

$$g_{<B}(i) = \frac{en}{B-i} + g_B(i).$$

For $t > 0$, let X_t be the parent bit string of (1+1) EA at iteration t. and let T be the iteration number at which (1+1) EA finds the optimum for the first time. Let

$$f(X_t) = \begin{cases} g_B(LO(X_t)) & \text{if } |X_t|_1 = B, \\ g_{<B}(LO(X_t) & \text{if } |X_t|_1 < B. \end{cases} \tag{1}$$

We consider two different cases, $|X_t|_1 = B$ and $|X_t|_1 < B$ and show in both the cases the drift is at least 1. Suppose we are in an iteration $t < T$ with $LO(X_t) = i$ and $|X_t|_1 = B$. Then the probability that the number of 1s in the search point can decrease by 1 in the next iteration is at least $\frac{B-i}{en}$. This is because we can get a desired search point by flipping only one of the 1 bits of $B - i$, excluding the leading 1s, and not flipping any other bit. Therefore,

$$\begin{aligned} E[f(X_{t+1}) - f(X_t) \,\big|\, & LO(X_t) = i,\ |X_t|_1 = B] \\ &\geq (g_{<B}(i) - g_B(i))\cdot Pr(|X_{t+1}|_1 < B) \\ &\geq \left(\frac{en}{B-i} + g_B(i) - g_B(i)\right)\cdot\left(\frac{B-i}{en}\right) = 1. \end{aligned}$$

Suppose we are in an iteration $t < T$ with $LO(X_t) = i$ and $|X_t|_1 < B$. Then in the next iteration the value of LeadingOnes can increase when the leftmost 0 is flipped to 1 as this does not violate the constraint. This happens with probability at least $\frac{1}{en}$. Since $|X_t|_1 < B$, we can also stay in the same level (same number of leading 1s) and the number of 1s can increase to B with probability at most $\frac{n-B}{n}$ (see Lemma 1). This implies that the potential can decrease by $\frac{en}{B-i}$ with probability at most $\frac{n-B}{n}$.

$$\begin{aligned} E[f(X_{t+1}) - & f(X_t) \mid LO(X_t) = i, |X_t|_1 < B] \\ &\geq (g(i+1, B) - g_{<B}(i))\cdot\frac{1}{en} - \left(\frac{en}{B-i}\cdot\frac{n-B}{n}\right) \\ &\geq \left(en\left(1 + \frac{e\cdot(n-B)}{B-i}\right) + g_{<B}(i) - g_{<B}(i)\right)\cdot\left(\frac{1}{en}\right) \\ &\quad - \left(\frac{e\cdot(n-B)}{B-i}\right) \\ &= 1. \end{aligned}$$

This results in an expected additive drift value greater than 1 in all the cases, so according to the additive drift theorem [10, Theorem 5],

$$\begin{aligned}
E[T] &\leq f(X_T) = g_B(B) \\
&= \sum_{i=0}^{B-1}(g_{<B}(i) - g_B(i)) + \sum_{i=1}^{B}(g_B(i) - g_{<B}(i-1)) \\
&= \sum_{i=0}^{B-1}\frac{en}{B-i} + \sum_{i=1}^{B} en\left(1 + \frac{e\cdot(n-B)}{B-i+1}\right) \\
&= en\sum_{i=1}^{B}\left(\frac{1}{i}\right) + enB + e^2\cdot n(n-B)\sum_{i=1}^{B}\left(\frac{1}{i}\right) \\
&\leq en(\log B + 1) + enB + e^2\cdot n(n-B)(\log B + 1) \\
&= O(n^2 + n(n-B)\log B).
\end{aligned}$$

□

We now turn to the lower bound. When (1+1) EA optimizes LeadingOnes in unconstrained setting the probability that a bit which is after the left-most 0 is 1 is exactly $\frac{1}{2}$. But this is not true in the constrained setting. The following lemma gives an upper bound on this probability during the cardinality constraint optimization.

Lemma 3. *For any $t \geq 0$, let x^t denote the search point at iteration t when (1+1) EA is optimizing LeadingOnes with the cardinality constraint B. Then for any $t \geq 0$ and $i > LO(x^t)$, $Pr(x_i^t = 1) \leq 1/2$.*

Proof. We will prove this by induction. The base case is true because we have an uniform random bit string at $t = 0$. Lets assume that the statement is true for t, i.e. for any $i > LO(x_t)$, $Pr(x_i^t = 1) \leq 1/2$. Let A be the event that the offspring is accepted. Then, for $i > LO(x_{t+1})$,

$$\begin{aligned}
Pr(x_i^{t+1} = 1) = Pr((x_i^t = 0) &\cap (i^{th} \text{ bit is flipped}) \cap A) \\
+ Pr((x_i^t = 1) &\cap (i^{th} \text{ bit is flipped}) \cap A^c) \\
+ Pr((x_i^t = 1) &\cap (i^{th} \text{ bit is not flipped}).
\end{aligned}$$

Let $Pr(x_i^t = 1) = p$, $Pr(A \mid (i^{th} \text{ bit is flipped} \cap x_i^t = 0)) = a$ and $Pr(A \mid (i^{th} \text{ bit is flipped} \cap x_i^t = 1)) = b$. Then note that $a \leq b$ (because we have at least as many events as in probability a contributing to the probability b) and by induction hypothesis,

$$\begin{aligned}
Pr(x_i^{t+1} = 1) &= (1-p)\cdot 1/n\cdot a + p\cdot 1/n\cdot(1-b) + p\cdot(1-1/n) \\
&= a/n - (p\cdot a)/n + p/n - (p\cdot b)/n + p - p/n \\
&= a/n - p\cdot(1 - a/n - b/n) \\
&\leq a/n + 1/2\cdot(1 - a/n - b/n) \\
&\leq a/n + 1/2\cdot(1 - a/n - a/n) = 1/2.
\end{aligned}$$

□

We use the previous lemma to prove the $\Omega(n^2)$ lower bound on the expected time in the next theorem.

Theorem 4. *Let $n, B \in \mathbb{N}$. Then the expected optimization time of the (1+1) EA on the LeadingOnes with cardinality constraint B is $\Omega\left(n^2\right)$.*

Proof. We use the *fitness level method with visit probabilities* technique defined in [3, Theorem 8] to prove this lower bound. Similar to [3, Theorem 11], we also partition the search space $\{0,1\}^n$ based on the LeadingOnes values. For every $i \leq B$, let A_i contain all the bit strings with the LeadingOnes value i. If our search point is in A_i, then we say that the search point is in the state i. For every $i \in \{1, \cdots, B-1\}$, we have to find the visit probabilities v_i and an upper bound for p_i, the probability to leave the state i.

The best case scenario for the search point to leave the state i is when the number of 1s in the search point is less than B. In this case, we have to flip the $(i+1)^{th}$ bit to 1 and should not flip any of the first i bits to 0. This happens with the probability $\frac{1}{n}\cdot\left(1-\frac{1}{n}\right)^i$. Therefore, for every $i \in \{1, \cdots, B-1\}$, $p_i \leq \frac{1}{n}\cdot\left(1-\frac{1}{n}\right)^i$.

Next, we claim that, for each $i \in \{1, \cdots, B-1\}$, v_i – the probability to visit the state i is at least $\frac{1}{2}$. We use [3, Lemma 10] to show this. Suppose the initial search point is in a state greater than or equal to i, then the probability for it to be in state i is equal to the probability that the $(i+1)^{th}$ bit is 0. Since the initial bit string is chosen uniformly at random the probability that the $(i+1)^{th}$ bit is 0 is $\frac{1}{2}$. This shows the first required bound on the probability for the lemma in [3, Lemma 10]. Suppose the search point is transitioning into a level greater than or equal to i, then the probability that it transition into state i is equal to the probability that $(i+1)^{th}$ bit is 0. From Lemma 3, we know that this probability is at least 1/2. This gives the second bound required for the [3, Lemma 10], therefore v_i is at least $\frac{1}{2}$.

By using fitness level method with visit probabilities theorem [3, Theorem 8], if T is the time taken by the (1+1) EA to find an individual with B number of LeadingOnes for the first time then, we have, $E[T] \geq \sum_{i=0}^{B-1}\frac{v_i}{p_i} = \frac{n}{2}\cdot\sum_{i=0}^{B-1}\left(1-\frac{1}{n}\right)^{-i} \geq \frac{n^2}{2} = \Omega(n^2)$. □

We aim to show the $\Omega(n^2 + n(n-B)\log B)$ lower bound and Theorem 4 gives the $\Omega(n^2)$ lower bound. Therefore, next we consider the case where B is such that $n(n-B)\log B \neq O(n^2)$ to prove the desired lower bound.

Theorem 5. *Let $n, B \in \mathbb{N}$ and suppose $n(n-B)\log B = \omega(n^2)$. Then the expected optimization time of the (1+1) EA on the objective LeadingOnes with cardinality constraint B is $\Omega\left(n(n-B)\log B\right)$.*

Proof. We consider the potential function g such that, for all $x \in \{0,1\}^n$,

$$g(x) = \frac{n\cdot|x|_1}{B - LO(x) + 1} + \sum_{i=LO(x)}^{B-1}\frac{n(n-B)}{32e^2(B-i)}.$$

The first term appreciates progress by reducing the number of 1s. This is scaled to later derive constant drift in expectation from such a reduction whenever $|x|_1 = B$, the case where progress by increasing the number of leading 1s is not easy. The second term appreciates progress by increasing the number of leading 1s, scaled to derive constant drift in case of $|x|_1 < B$.

The idea of the proof is as follows. We show that the potential decreases by at most 10 in expectation. Then the lower bound of additive drift theorem will give the desired lower bound on the expected run time (see [10, Theorem 5]).

We start by calculating the expected potential at $t = 0$. Since the initial bit string is chosen uniformly at random the probability that the first bit is 0 is $\frac{1}{2}$. Therefore $Pr(LO(x_0) = 0) = \frac{1}{2}$, which implies

$$\begin{aligned} E[g(x_0)] &\geq \frac{1}{2} \cdot E\left[\sum_{i=LO(x_0)}^{B-1} \frac{n(n-B)}{32e^2(B-i)} \;\Big|\; LO(x_0) = 0\right] \\ &= \frac{n(n-B)}{64e^2} \sum_{i=0}^{B-1} \frac{1}{B-i} \\ &= \frac{n(n-B)}{64e^2} \cdot \sum_{i=1}^{B} \frac{1}{i} \geq \frac{n(n-B)\ln(B)}{64e^2}. \end{aligned}$$

Therefore, there exits a constant $c > 0$ such that $E[g(x_0)] \geq cn(n-B)\log B$. The optimum has a potential value of nB; thus, we can find a lower bound on the optimization time by considering the time to find a potential value of at most nB. Let $T = \min\{t \geq 0 \mid g(x_t) \leq nB\}$. Note that T may not be the time at which we find the optimum for the first time. From $n(n-B)\log B = \omega(n^2)$ we get, for n large enough, that $E[g(x_0)] > nB$, which implies that the expected optimization time is at least $E[T]$.

In order to show the lower bound on the drift, we consider two different cases, $|x_t|_1 = B$ and $|x_t|_1 < B$ and show in both the cases drift is at most 10. First, we examine the case where the algorithm has currently B number of 1s. For any t, let A_t be the event that $|x_t|_1 = B$ and let $\Delta_t = g(x_t) - g(x_{t+1})$ and

$$\begin{aligned} \Delta_t^s &= \sum_{i=LO(x_t)}^{B-1} \frac{n(n-B)}{32e^2(B-i)} - \sum_{i=LO(x_{t+1})}^{B-1} \frac{n(n-B)}{32e^2(B-i)} \\ &= \sum_{i=LO(x_t)}^{LO(x_{t+1})-1} \frac{n(n-B)}{32e^2(B-i)}. \end{aligned}$$

Then, $$\begin{aligned} E[\Delta_t \mid A_t] &= n \cdot E\left[\frac{|x_t|_1}{B - LO(x_t) + 1} - \frac{|x_{t+1}|_1}{B - LO(x_{t+1}) + 1} \;\Big|\; A_t\right] \\ &\quad + E[\Delta_t^s \mid A_t] \\ &\leq n \cdot E\left[\frac{|x_t|_1 - |x_{t+1}|_1}{B - LO(x_{t+1}) + 1} \;\Big|\; A_t\right] + E[\Delta_t^s \mid A_t]. \end{aligned}$$

Now we calculate the bounds for all the required expectations in the above equation.

First we calculate a bound for $n \cdot E\left[\frac{|x_t|_1 - |x_{t+1}|_1}{B - LO(x_{t+1}) + 1} \mid A_t\right]$ by using the definition of the expectation. Let $I = \{0, \cdots, B - LO(x_t)\}$ and $J = \{-1, 0, \cdots, B - LO(x_{t+1}) - 1\}$. Then the possible values the random variable $|x_t|_1 - |x_{t+1}|_1$ can have are the values in I. And the possible values $B - LO(x_{t+1}) + 1$ can have are $\{B - LO(x_t) - j \mid j \in J\}$. For $i \in \{1, \cdots, B - LO(x_t)\}$, the probability $Pr((|x_t|_1 - |x_{t+1}|_1 = i) \cap (B - LO(x_{t+1}) + 1 = B - LO(x_t) + 1)) \leq \binom{B - LO(x_t)}{i} \left(\frac{1}{n}\right)^i \leq \frac{B - LO(x_t)}{i!n}$ and for $i \in \{1, \cdots, B - LO(x_t)\}$ and $j \in \{0, \cdots, B - LO(x_{t+1}) - 1\}\}$, the probability $Pr((|x_t|_1 - |x_{t+1}|_1 = i) \cap (B - LO(x_{t+1}) + 1 = B - LO(x_t) - j)) \leq \binom{B - LO(x_t)}{i} \left(\frac{1}{n}\right)^{i+1} \frac{1}{2^j} \leq \frac{B - LO(x_t)}{i!n^2} \frac{1}{2^j}$ (see Lemma 3). For $i \in I$ and $j \in J$, let $p_t^{ij} = Pr((|x_t|_1 - |x_{t+1}|_1 = i) \cap (B - LO(x_{t+1}) + 1 = B - LO(x_t) - j))$ and $K = J \setminus \{B - LO(x_t) + 1\}$ and $K^c = \{B - LO(x_t) + 1\}$. Then, $n \cdot E\left[\frac{|x_t|_1 - |x_{t+1}|_1}{B - LO(x_{t+1}) + 1} \mid A_t\right]$

$$\begin{aligned} &= n \cdot \sum_{i \in I} \sum_{j \in J} \frac{i \cdot p_t^{ij}}{B - LO(x_t) - j} \\ &= n \cdot \sum_{i \in I} \sum_{j \in K^c} \frac{i \cdot p_t^{ij}}{B - LO(x_t) - j} + n \cdot \sum_{i \in I} \sum_{j \in K} \frac{i \cdot p_t^{ij}}{B - LO(x_t) - j} \\ &= \sum_{i \in I} \frac{i(B - LO(x_t))}{i!(B - LO(x_t) + 1)} + \sum_{i \in I} \frac{i(B - LO(x_t))}{i!n} \sum_{j \in K} \frac{\frac{1}{2^j}}{B - LO(x_t) - j} \\ &\leq \sum_{i \in I \setminus \{0\}} \frac{1}{(i-1)!} + \sum_{i \in I} \frac{1}{(i-1)!} \sum_{j \in K} \frac{1}{2^j} \\ &\leq e + 2e = 3e \leq 9. \end{aligned} \tag{2}$$

We used the infinite sum values $\sum_{i=1}^{\infty} \frac{1}{(i-1)!} = e$, $\sum_{i=0}^{\infty} \frac{1}{2^i} = 2$, to bound our required finite sums in the above calculation.

Now, we calculate $E[\Delta_t^s \mid A_t]$, to get an upper bound for $E[\Delta_t \mid A_t]$. When $|x_t|_1 = B$, the probability to gain in the LeadingOnes-values is at most $\frac{B - LO(x_t)}{n} \cdot \frac{1}{n}$. Therefore we have

$$\begin{aligned} E[\Delta_t^s \mid A_t] &= \frac{n(n-B)}{32e^2} \cdot E\left[\sum_{i=LO(x_t)}^{LO(x_{t+1})-1} \frac{1}{B-i} \;\Big|\; A_t\right] \\ &\leq \frac{n(n-B)}{32e^2} \cdot E\left[\frac{LO(x_{t+1}) - LO(x_t)}{B - LO(x_{t+1}) + 1} \;\Big|\; A_t\right]. \end{aligned} \tag{3}$$

We calculate an upper bound for $E\left[\frac{LO(x_{t+1}) - LO(x_t)}{B - LO(x_{t+1}) + 1} \mid A_t\right]$. The probability that $LO(x_{t+1}) - LO(x_t) = i$ given that we gain at least a leading one is the probability that next $i - 1$ bits after left-most 0 bit) is 1 followed by a 0 bit. This implies that the probability that $LO(x_{t+1}) - LO(x_t) = i$ given that we gain at least a leading one is at most $\frac{1}{2^{i-1}}$. Therefore, we have $E\left[\frac{LO(x_{t+1}) - LO(x_t)}{B - LO(x_{t+1}) + 1} \mid A_t\right]$

$$\leq \frac{B - LO(x_t)}{n} \cdot \frac{1}{n} \cdot \sum_{i=1}^{B - LO(x_t) - 1} \frac{i \cdot 2^{1-i}}{B - LO(x_t) - i}. \tag{4}$$

Equations 3 and 4 imply that, $E[\Delta_t^s \mid A_t]$

$$\begin{aligned} &\leq \frac{n(n-B)}{32e^2} \cdot \frac{B - LO(x_t)}{n} \cdot \frac{1}{n} \cdot \sum_{i=1}^{B - LO(x_t) - 1} \frac{i \cdot 2^{1-i}}{B - LO(x_t) - i} \\ &\leq \frac{1}{16e^2} \cdot \sum_{i=1}^{B - LO(x_t) - 1} \frac{(B - LO(x_t)) \cdot i}{(B - LO(x_t) - i) \cdot 2^i} \\ &= \frac{1}{16e^2} \cdot \sum_{i=1}^{B - LO(x_t) - 1} \frac{(B - LO(x_t) - i + i) \cdot i}{(B - LO(x_t) - i) \cdot 2^i} \\ &\leq \frac{1}{16e^2} \left(\sum_{i=1}^{B - LO(x_t) - 1} \frac{i}{2^i} + \sum_{i=1}^{B - LO(x_t) - 1} \frac{i^2}{2^i} \right) \\ &\leq \frac{1}{16e^2}(2 + 6) = \frac{1}{2e^2} \leq 1. \end{aligned} \tag{5}$$

We used the infinite sum values $\sum_{i=1}^{\infty} \frac{i}{2^i} = 2$, $\sum_{i=0}^{\infty} \frac{i^2}{2^i} = 6$, to bound our required finite sums in the above calculation.

From Equations 2 and 5, we have $E[\Delta_t \mid A_t] \leq 10$ which concludes the first case (when $|x_t|_1 = B$). Next we calculate the bound for the drift conditioned on the event A_t^c (when $|x_t|_1 < B$).

$$\begin{aligned}
E[\Delta_t \mid A_t^c] &= n \cdot E\left[\frac{|x_t|_1}{B - LO(x_t) + 1} - \frac{|x_{t+1}|_1}{B - LO(x_{t+1}) + 1} \mid A_t^c\right] \\
&\quad + E[\Delta_t^s \mid A_t^c] \\
&\leq n \cdot E\left[\frac{|x_t|_1 - |x_{t+1}|_1}{B - LO(x_{t+1}) + 1} \mid A_t^c\right] + E[\Delta_t^s \mid A_t^c].
\end{aligned}$$

Similar to the previous case, for this case also we start by finding a bound for $n \cdot E\left[\frac{|x_t|_1 - |x_{t+1}|_1}{B - LO(x_{t+1})+1} \mid A_t^c\right]$. Let $\Delta_t^1 = |x_t|_1 - |x_{t+1}|_1$. Then $n \cdot E\left[\frac{\Delta_t^1}{B-LO(x_{t+1})+1} \mid A_t^c\right]$

$$\begin{aligned}
&= n \cdot E\left[\frac{\Delta_t^1}{B - LO(x_{t+1}) + 1} \mid A_t^c, \Delta_t^1 > 0\right] \cdot Pr(\Delta_t^1 > 0) \\
&\quad + n \cdot E\left[\frac{\Delta_t^1}{B - LO(x_{t+1}) + 1} \mid A_t^c, \Delta_t^1 < 0\right] \cdot Pr(\Delta_t^1 < 0).
\end{aligned}$$

Now we find upper bounds for both the quantities in the above equation. By doing calculations similar to the calculations which lead to the Equation (2), we get $n \cdot E\left[\frac{\Delta_t^1}{B-LO(x_{t+1})+1} \mid A_t^c, \Delta_t^1 > 0\right] \leq 9$. Since there are at least $n - B$ number of 0 bits, the probability to gain a 1 bit is at least $\frac{n-B}{en}$. And the probability that $LO(x_t) = LO(x_{t+1})$ is at least $\frac{1}{2e}$, for n large enough. Therefore, $n \cdot E\left[\frac{\Delta_t^1}{B-LO(x_{t+1})+1} \mid \Delta_t^1 < 0\right] \cdot Pr(\Delta_t^1 < 0) \leq -\frac{(n-B)}{2e^2(B-LO(x_t)+1)}$. By combining these two bounds we have

$$n \cdot E\left[\frac{\Delta_t^1}{B - LO(x_{t+1}) + 1} \mid A_t^c\right] \leq 9 - \frac{(n-B)}{2e^2(B - LO(x_t) + 1)}. \quad (6)$$

Next we calculate $E[\Delta_t^s \mid A_t^c]$, to get an upper bound for $E[\Delta_t \mid A_t^c]$. When $|x_t|_1 < B$, the probability to gain in LeadingOnes-value is at most $\frac{1}{n}$. Therefore,

$$\begin{aligned}
E[\Delta_t^s \mid A_t^c] &\leq \frac{n(n-B)}{32e^2} \cdot E\left[\sum_{i=LO(x_t)}^{LO(x_{t+1})-1} \frac{1}{B-i} \mid A_t^c\right] \\
&\leq \frac{n(n-B)}{32e^2} \cdot E\left[\frac{LO(x_{t+1}) - LO(x_t)}{B - LO(x_{t+1}) + 1} \mid A_t^c\right] \\
&\leq \frac{n(n-B)}{32e^2} \cdot \frac{1}{n} \cdot \sum_{i=1}^{B-LO(x_t)-1} \frac{i}{(B - LO(x_t) - i) \cdot 2^{i-1}} \\
&= \frac{n-B}{32e^2(B - LO(x_t))} \cdot \sum_{i=1}^{B-LO(x_t)-1} \frac{(B - LO(x_t)) \cdot i}{(B - LO(x_t) - i) \cdot 2^{i-1}} \\
&= \frac{n-B}{16e^2(B - LO(x_t))} \cdot \sum_{i=1}^{B-LO(x_t)-1} \frac{(B - LO(x_t) - i + i) \cdot i}{(B - LO(x_t) - i) \cdot 2^{i}} \\
&\leq \frac{n-B}{16e^2(B - LO(x_t))} \left(\sum_{i=1}^{B-LO(x_t)-1} \frac{i}{2^i} + \sum_{i=1}^{B-LO(x_t)-1} \frac{i^2}{2^i}\right) \\
&\leq \frac{n-B}{16e^2(B - LO(x_t))}(2+6) = \frac{n-B}{2e^2(B - LO(x_t))}. \quad (7)
\end{aligned}$$

Since $B - LO(x_t) \geq 1$, we have $\frac{n-B}{2e^2(B-LO(x_t))} \leq \frac{n-B}{e^2(B-LO(x_t)+1)}$. From Equations 6 and 7,we have

$$E[\Delta_t \mid A_t] \leq 9 - \frac{(n-B)}{2e^2(B - LO(x_t) + 1)} + \frac{n-B}{2e^2(B - LO(x_t))} \leq 10.$$

Which concludes the second case (when $|x_t|_1 < B$). Now we have $E[\Delta_t \mid g(x_t)] \leq 10$. Therefore, by the lower bounding additive drift theorem [10, Theorem 5],

$$E[T] \geq \frac{E[g(x_0)] - nB}{10} = \Omega(n(n-B)\log B).$$

□

Corollary 6. *Let $n, B \in \mathbb{N}$. Then the expected optimization time of the (1+1) EA on the LeadingOnes with cardinality constraint B is $\Theta\left(n^2 + n(n-B)\log B\right)$.*

Proof. From Theorem 4 and Theorem 5 we have the required lower bound and we have the upper bound from Theorem 2. Therefore the expected optimization time is $\Theta\left(n^2 + n(n-B)\log B\right)$. □

4 BETTER RUN TIMES

In this section we discuss two ways to obtain the (optimal) run time of $O(n^2)$. First, we state a corollary to the proof of Theorem 2, that we can almost reach the bound within $O(n^2)$ iterations.

Corollary 7. *Let $n, B \in \mathbb{N}$ and $c > 0$. Then the (1+1) EA on LeadingOnes with the cardinality constraint B finds a search point with $B - c(n-B)$ leading 1s within $O(n^2)$ in expectation.*

With the next theorem we show that incorporating the number of 0s of a bit string as a secondary objective gives an expected run time of the (1+1) EA of $\Theta(n^2)$ to optimize cardinality constrained LeadingOnes.

Theorem 8. *Let $B \leq n-1$ and for any $x \in \{0,1\}^n$, let*

$$f(x) = \begin{cases} (LO(x), |x|_0) & |x|_1 \leq B, \\ -|x|_1 & \textit{otherwise}. \end{cases}$$

Then (1+1) EA takes $\Theta(n^2)$ in expectation to optimize f in the lexicographic order with the cardinality constraint B.

Proof. For any $x \in \{0,1\}^n$, let $g(x) = 3eLO(x) + |x|_0$, where $|x|_0$ represents the number of 0s in x. Intuitively, we value both progress in decreasing the number of (unused) 1s, as well as an increase in leading 1s, but we value an increase in leading 1s higher (since this is the ultimate goal, and typically comes at the cost of increasing the number of 1 by a constant). Now we will show that $g(y) = 3eB + n - B$ if and only if y is the optimum of f. Suppose for some $y \in \{0,1\}^n$, $g(y) = 3eB + n - B$. Then $3eLO(y) + |y|_0 = 3eB + n - B$, which implies that $3eLO(y) = 3eB + n - B - |y|_0$. Since $LO(y) \leq B$ and $|y|_0 \leq n - LO(y)$, $3eLO(y) = 3eB + n - B - |y|_0$ implies that $LO(y) = B$. Therefore, y is optimal.

Let $T = \min\{t \geq 0 \mid g(x_t) \geq 3eB + n - B\}$. We will examine the drift at two different scenarios, $|x_t|_1 < B$ and $|x_t|_1 = B$ and show that in both the cases the drift is at least $1/n$. Let $\Delta_t = g(x_{t+1}) - g(x_t)$ and A_t be the event that the left-most 0 in x_t is flipped. Then

$E[\Delta_t \mid A_t^c] \geq 0$, because, if the number of LeadingOnes does not increase then $|x_{t+1}|_0 - |x_t|_0 \geq 0$ which in turn implies $\Delta_t \geq 0$. Therefore, for any $0 \leq t < T$,

$$\begin{aligned}E[\Delta_t \mid |x_t|_1 < B] &= E[\Delta_t \mid A_t, |x_t|_1 < B] \cdot Pr[A_t] \\ &\quad + E[\Delta_t \mid A_t^c, |x_t|_1 < B] \cdot Pr[A_t^c] \\ &\geq \frac{1}{n} \cdot E[g(x_{t+1}) - g(x_t) \mid A_t, |x_t|_1 < B] + 0 \\ &= \frac{1}{n} \cdot 3eE[LO(x_{t+1}) - LO(x_t) \mid A_t, |x_t|_1 < B] \\ &\quad + \frac{1}{n} \cdot E[|x_{t+1}|_0 - |x_t|_0 \mid A_t, |x_t|_1 < B].\end{aligned}$$

Note that $E[LO(x_{t+1}) - LO(x_t) \mid A_t, |x_t|_1 < B]$ is greater than or equal to the probability of not flipping any other bits, since it increases the number of LeadingOnes by at least one. And $E[|x_t|_0 - |x_{t+1}|_0 \mid A_t, |x_t|_1 < B])$ is upper bounded by the sum $1 + \sum_{i=1}^{|x_t|_0 - 1} Pr(\text{flipping the } i^{th} \text{ 0 bit})$. This is because we lose one 0 bit by flipping the left-most 0 bit and we flip each other 0-bit independently with probability $\frac{1}{n}$. And $\frac{|x_t|_0 - 1}{n} \leq 1$, therefore,

$$E[\Delta_t \mid |x_t|_1 < B] \geq \frac{1}{n}\left(3e\left(1 - \frac{1}{n}\right)^{n-1} - \left(1 + \frac{|x_t|_0 - 1}{n}\right)\right) \geq \frac{1}{n}.$$

This concludes the first case. Now, lets consider the case $|x_t|_1 = B$. Let D be the event that the mutation operator flips exactly one 1 bit which lies after the left-most 0 bit and flips no other bits. Since $|x_t|_1 = B$ and $LO(x_t) < B$, there is at least one such 1 bit, which implies $E[|x_{t+1}|_0 - |x_t|_0 \mid |x_t|_1 = B, D] \geq 1$. Also note that $Pr(D) \geq \frac{1}{en}$. If a search point is accepted, then the number of 1 bits is at most B and the LeadingOnes value cannot decrease; thus, $LO(x_{t+1}) \geq LO(x_t)$ and $|x_{t+1}|_0 \geq n - B$. Overall we have $g(x_{t+1}) = 3eLO(x_{t+1}) + |x_{t+1}|_0 \geq 3eLO(x_t) + n - B = g(x_t)$. Therefore, $E[\Delta_t \mid |x_t|_1 = B, D^c] \geq 0$ and

$$\begin{aligned}E[\Delta_t \mid |x_t|_1 = B] &= E[|x_{t+1}|_0 - |x_t|_0 \mid |x_t|_1 = B, D] \cdot Pr(D) \\ &\quad + E[\Delta_t \mid |x_t|_1 = B, D^c] \cdot Pr(D^c) \\ &\geq \frac{1}{en}.\end{aligned}$$

The expected number of 0s in the initially selected uniform random bit string is $\frac{n}{2}$ and the expected number of LeadingOnes is at least zero, therefore $E[g(x_0)] \geq \frac{n}{2}$. We have an drift of at least $\frac{1}{en}$ in both the cases, therefore we get the required upper bound by the additive drift theorem [10, Theorem 5],

$$E[T] \leq en \cdot (3eB + n - B - E[g(x_0)]) \leq 3e^2nB + \frac{en^2}{2} - enB = O(n^2).$$

This proves the upper bound. And the lower bound follows from Theorem 4. □

5 EMPIRICAL ANALYSIS

We want to extend our theoretical work on deterministic constraint the case of stochastic constraint models (as defined in Section 2.2). For the first model we use parameters $\mu = 1$ and $\sigma = 0.1$ and for the second model we use $\epsilon = \sqrt{3}$. Note that in the second model $U(B - \sqrt{3}, B + \sqrt{3})$ has variance 1. For both the models we considered two different B values 75 and 95 (also $B = 85$ in the Appendix). As we will see, the (1+1) EA struggles in these settings; in order to show that already a small parent population can remedy this, we also consider the $(10+1)$ EA in our experiments.

We use the following lemma for discussing certain probabilities in this section.

Lemma 9. *Let $k \in \{1, \cdots, B-1\}$, $x \in \{0,1\}^n$, $B \in [n]$, $W_x = \sum_{i=1}^n x_i \cdot Y_i$ where $Y_i \sim N(1, \sigma^2)$ and x_i be the i–th bit of x and $|x|_1 \leq B - k$. Then $Pr(W_x > B) \leq \frac{1}{\sqrt{\pi}} e^{\frac{-k^2}{2n^2\sigma^2}}$ and $Pr(W_x > B \mid |x|_1 = B) = \frac{1}{2}$.*

In Figure 1 we have a single sample run of (1+1) EA on the first model. We observe that if the (1+1) EA finds a bit string with B number of 1s it violates the constraint with probability $\frac{1}{2}$ (see Lemma 9) and accepts a bit string with a lower number of LeadingOnes. This process keeps repeating whenever the (1+1) EA encounters an individual with a number of 1s closer to B.

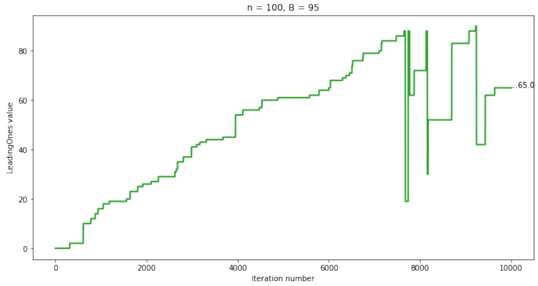

Figure 1: (1+1) EA sample run with $n = 100$, $B = 85$ and $N(1, 0.1)$ chance constraint for 10000 iteration.

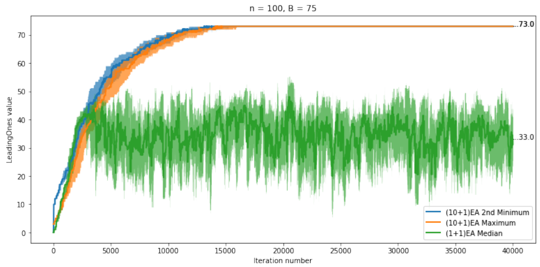

Figure 2: (10+1) EA and (1+1) EA on LeadingOnes with $n = 100$, $B = 75$ and $N(1, 0.1)$ chance constraint for 40000 iterations.

Figures 2 and 3 are about the first model in which we have the LeadingOnes-values of the best individual (bit string with the maximum fitness value) in each iteration of the (10+1) EA, the LeadingOnes values of the second-worst individuals (bit string with the second-smallest fitness value) in each iteration of the (10+1) EA and the LeadingOnes values at each iteration of the (1+1) EA. Each curve is the median of thirty independent runs and the shaded area is the area between the 25−th and the 75−th quantile values. For all three B-values, after initial iterations, all the individuals except the worst individual in the (10+1) EA population have $B - 2$ number of leading 1s. This is because, for this model, the probability that an individual with $B - 2$ number of 1s violates the constraint is at most $\frac{e^{-2}}{\sqrt{\pi}}$ (from Lemma 9).

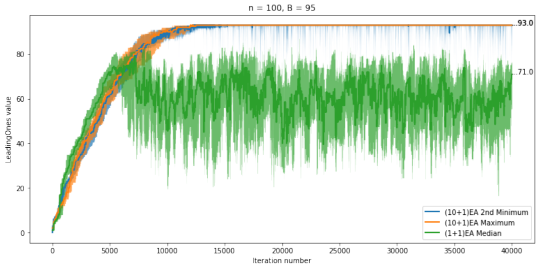

Figure 3: (10+1) EA and (1+1) EA on LeadingOnes with $n = 100$, $B = 95$ and $N(1, 0.1)$ chance constraint for 40000 iterations.

Figures 4 and 5 are about the second model and the curves represent the same things as in the previous figures but with respect to the second model. In these figures we can see that the best and the second worst individuals of the (10+1) EA are not the same because of the changing constraint values.

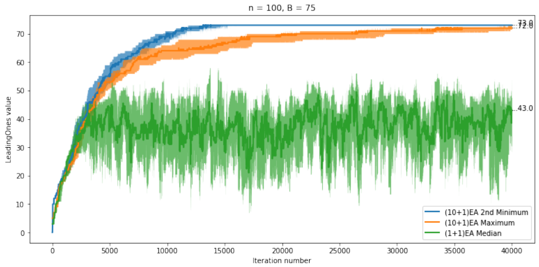

Figure 4: (10+1) EA and (1+1) EA on LeadingOnes with $n = 100$, $B = 75$ and $U(B - \sqrt{3}, B + \sqrt{3})$ stochastic constraint for 40000 iterations.

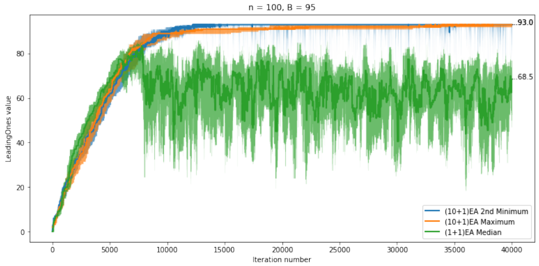

Figure 5: (10+1) EA and (1+1) EA on LeadingOnes with $n = 100$, $B = 95$ and $U(B - \sqrt{3}, B + \sqrt{3})$ stochastic constraint for 40000 iterations.

6 CONCLUSIONS

Understanding how evolutionary algorithms deal with constrained problems is an important topic of research. We investigated the classical LeadingOnes problem with additional constraints. For the case of a deterministic uniform constraint we have carried out a rigorous run time analysis of the (1+1) EA which gives results on the expected optimization time in dependence of the chosen constraint bound. Afterwards, we examined stochastic constraints and the use of larger populations for dealing with uncertainties. Our results show a clear benefit of using the $(10 + 1)$ EA instead of the $(1 + 1)$ EA. We regard the run time analysis of population-based algorithms for our examined settings of stochastic constraints as an important topic for future work.

7 ACKNOWLEDGEMENTS

Frank Neumann has been supported by the Australian Research Council (ARC) through grant FT200100536. Tobias Friedrich and Timo Kötzing were supported by the German Research Foundation (DFG) through grant FR 2988/17-1.

REFERENCES

[1] Hans-Georg Beyer and Bernhard Sendhoff. 2007. Robust optimization–a comprehensive survey. *Computer methods in applied mechanics and engineering* 196, 33-34 (2007), 3190–3218.

[2] Abraham Charnes and William W Cooper. 1959. Chance-constrained programming. *Management science* 6, 1 (1959), 73–79.

[3] Benjamin Doerr and Timo Kötzing. 2021. Lower Bounds from Fitness Levels Made Easy. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2021)*. Association for Computing Machinery, New York, NY, USA, 1142–1150. https://doi.org/10.1145/3449639.3459352

[4] Benjamin Doerr and Frank Neumann (Eds.). 2020. *Theory of Evolutionary Computation - Recent Developments in Discrete Optimization*. Springer. https://doi.org/10.1007/978-3-030-29414-4

[5] Stefan Droste, Thomas Jansen, and Ingo Wegener. 2002. On the analysis of the (1+1) evolutionary algorithm. *Theor. Comput. Sci.* 276, 1-2 (2002), 51–81.

[6] A. E. Eiben and James E. Smith. 2015. *Introduction to Evolutionary Computing, Second Edition*. Springer.

[7] Tobias Friedrich, Timo Kötzing, J. A. Gregor Lagodzinski, Frank Neumann, and Martin Schirneck. 2020. Analysis of the (1+1) EA on subclasses of linear functions under uniform and linear constraints. *Theor. Comput. Sci.* 832 (2020), 3–19.

[8] Tobias Friedrich, Timo Kötzing, J. A. Gregor Lagodzinski, Frank Neumann, and Martin Schirneck. 2017. Analysis of the (1+1) EA on Subclasses of Linear Functions under Uniform and Linear Constraints. In *Foundations of Genetic Algorithms (FOGA)*. ACM Press, 45–54.

[9] Grani A Hanasusanto, Vladimir Roitch, Daniel Kuhn, and Wolfram Wiesemann. 2015. A distributionally robust perspective on uncertainty quantification and chance constrained programming. *Mathematical Programming* 151 (2015), 35–62.

[10] Jun He and Xin Yao. 2004. A study of drift analysis for estimating computation time of evolutionary algorithms. *Natural Computing* 3 (2004), 21–35.

[11] Thomas Jansen. 2013. *Analyzing Evolutionary Algorithms - The Computer Science Perspective*. Springer. https://doi.org/10.1007/978-3-642-17339-4

[12] Pu Li, Harvey Arellano-Garcia, and Günter Wozny. 2008. Chance constrained programming approach to process optimization under uncertainty. *Computers & chemical engineering* 32, 1-2 (2008), 25–45.

[13] Bruce L Miller and Harvey M Wagner. 1965. Chance constrained programming with joint constraints. *Operations Research* 13, 6 (1965), 930–945.

[14] Rahul Nair and Elise Miller-Hooks. 2011. Fleet management for vehicle sharing operations. *Transportation Science* 45, 4 (2011), 524–540.

[15] Aneta Neumann and Frank Neumann. 2020. Optimising Monotone Chance-Constrained Submodular Functions Using Evolutionary Multi-objective Algorithms. In *PPSN (1) (Lecture Notes in Computer Science, Vol. 12269)*. Springer, 404–417.

[16] Aneta Neumann, Yue Xie, and Frank Neumann. 2022. Evolutionary Algorithms for Limiting the Effect of Uncertainty for the Knapsack Problem with Stochastic Profits. In *Parallel Problem Solving from Nature - PPSN XVII - 17th International Conference, PPSN 2022, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 13398)*. Springer, 294–307. https://doi.org/10.1007/978-3-031-14714-2_21

[17] Frank Neumann, Mojgan Pourhassan, and Carsten Witt. 2021. Improved Runtime Results for Simple Randomised Search Heuristics on Linear Functions with a

Uniform Constraint. *Algorithmica* 83, 10 (2021), 3209–3237.

[18] Frank Neumann and Andrew M. Sutton. 2019. Runtime analysis of the (1 + 1) evolutionary algorithm for the chance-constrained knapsack problem. In *Proceedings of the 15th ACM/SIGEVO Conference on Foundations of Genetic Algorithms, FOGA 2019*. ACM, 147–153. https://doi.org/10.1145/3299904.3340315

[19] Frank Neumann and Carsten Witt. 2010. *Bioinspired Computation in Combinatorial Optimization.* Springer. https://doi.org/10.1007/978-3-642-16544-3

[20] Chao Qian, Jing-Cheng Shi, Yang Yu, and Ke Tang. 2017. On Subset Selection with General Cost Constraints. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017.* ijcai.org, 2613–2619. https://doi.org/10.24963/ijcai.2017/364

[21] Chao Qian, Yang Yu, and Zhi-Hua Zhou. 2015. Subset Selection by Pareto Optimization. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015.* 1774–1782.

[22] Vahid Roostapour, Aneta Neumann, and Frank Neumann. 2022. Single- and multi-objective evolutionary algorithms for the knapsack problem with dynamically changing constraints. *Theor. Comput. Sci.* 924 (2022), 129–147. https://doi.org/10.1016/j.tcs.2022.05.008

[23] Vahid Roostapour, Aneta Neumann, Frank Neumann, and Tobias Friedrich. 2022. Pareto optimization for subset selection with dynamic cost constraints. *Artif. Intell.* 302 (2022), 103597. https://doi.org/10.1016/j.artint.2021.103597

[24] Feng Shi, Xiankun Yan, and Frank Neumann. 2022. Runtime Analysis of Simple Evolutionary Algorithms for the Chance-Constrained Makespan Scheduling Problem. In *PPSN (2) (Lecture Notes in Computer Science, Vol. 13399)*. Springer, 526–541.

[25] Eric W. Weisstein. [n. d.]. Complementary Error Function. https://mathworld.wolfram.com/Erfc.html. Accessed: 2023-02-01.

[26] Yue Xie, Oscar Harper, Hirad Assimi, Aneta Neumann, and Frank Neumann. 2019. Evolutionary algorithms for the chance-constrained knapsack problem. In *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO 2019)*. ACM, 338–346. https://doi.org/10.1145/3321707.3321869

[27] Yue Xie, Aneta Neumann, and Frank Neumann. 2020. Specific single- and multi-objective evolutionary algorithms for the chance-constrained knapsack problem. In *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO 2020)*. ACM, 271–279. https://doi.org/10.1145/3377930.3390162

[28] Yue Xie, Aneta Neumann, Frank Neumann, and Andrew M. Sutton. 2021. Runtime analysis of RLS and the (1+1) EA for the chance-constrained knapsack problem with correlated uniform weights. In *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO 2021)*. ACM, 1187–1194.

[29] Hui Zhang and Pu Li. 2011. Chance constrained programming for optimal power flow under uncertainty. *IEEE Transactions on Power Systems* 26, 4 (2011), 2417–2424.

A APPENDIX

Corollary 10. *Let $n, B \in \mathbb{N}$ and $c > 0$. Then the (1+1) EA on LeadingOnes with the cardinality constraint B finds a search point with $B - c(n-B)$ leading 1s within $O(n^2)$ in expectation.*

Proof. Let x_t be the search point at the t^{th} iteration of the (1+1) EA optimizing LeadingOnes with the cardinality constraint B and $T_1 = \min\{t \geq 0 \mid LO(x_t) \geq B - c(n-B)\}$. And let $k = B - c(n-B)$. Then, from the proof of Theorem 2 we know that

$$E[T_1] \leq g_{<B}(k) \leq \frac{enB}{2} + (en + e^2 n(n-B)) \left(1 + \ln\left(\frac{B}{c(n-B)}\right)\right).$$

Since $\ln y \leq y$ for any $y > 0$, we have $(n-B)\left(\ln\left(\frac{B}{c(n-B)}\right)\right) \leq \frac{B}{c} = O(n)$. Therefore, $E[T_1] = O(n^2)$. □

Corollary 11. *Let $n, B \in \mathbb{N}$ and $c > 0$. Then the (1+1) EA on LeadingOnes with the cardinality constraint B finds a search point with $B - c(n-B)$ number of LeadingOnes within $O(n^2)$ in expectation.*

Proof. Let x_t be the search point at the t^{th} iteration of the (1+1) EA optimizing LeadingOnes with the cardinality constraint B and $T_1 = \min\{t \geq 0 \mid LO(x_t) = B - c(n-B)\}$. And let $k = B - c(n-B)$. Then from Theorem 2 we know that,

$$\begin{aligned} E[T_1] &\leq g_{<B}(k) \\ &= \sum_{i=0}^{k} (g_{<B}(i) - g_B(i)) + \sum_{i=1}^{k} (g_B(i) - g_{<B}(i-1)) \\ &= \sum_{i=0}^{B-c(n-B)} \frac{en}{B-i} + \sum_{i=1}^{B-c(n-B)} en\left(1 + \frac{e \cdot (n-B)}{B-i+1}\right) \\ &\leq en \sum_{i=c(n-B)}^{B} \left(\frac{1}{i}\right) + \frac{enB}{2} + e^2 \cdot n(n-B) \sum_{i=c(n-B)}^{B} \left(\frac{1}{i}\right) \\ &\leq \frac{enB}{2} + (en + e^2 n(n-B)) \left(1 + \ln\left(\frac{B}{c(n-B)}\right)\right). \end{aligned}$$

Since $\ln y \leq y$ for any $y > 0$, we have $(n-B)\left(\ln\left(\frac{B}{c(n-B)}\right)\right) \leq \frac{B}{c} \leq n$. Therefore, $E[T_1] = O(n^2)$. □

Lemma 12. *Let $k \in \{1, \cdots, B-1\}$, $x \in \{0,1\}^n$, $B \in [n]$, $W_x = \sum_{i=1}^{n} x_i \cdot Y_i$ where $Y_i \sim N(1, \sigma^2)$ and x_i be the $i-$th bit of x and $|x|_1 \leq B - k$. Then $Pr(W_x > B) \leq \frac{1}{\sqrt{\pi}} e^{\frac{-k^2}{2n^2\sigma^2}}$.*

Proof. First note that W_x is nothing but sum of $|x|_1$ normal random variables with mean 1 and variance σ^2, i.e. $W_x \sim N(|x|_1, |x|_1 \cdot \sigma^2)$ and $Pr(W_x > B \mid |x|_1 = B) = \frac{1}{2}$.

$$\begin{aligned} Pr(W_x > B) &= 1 - Pr(W_x \leq B) \\ &= 1 - \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{B - |x|_1}{|x|_1 \cdot \sigma\sqrt{2}}\right)\right) \\ &= \frac{1}{2}\left(1 - \operatorname{erf}\left(\frac{B - |x|_1}{|x|_1 \cdot \sigma\sqrt{2}}\right)\right) \\ &= \frac{1}{2} \operatorname{erfc}\left(\frac{B - |x|_1}{|x|_1 \cdot \sigma\sqrt{2}}\right). \end{aligned}$$

Since complementary error function erfc is a decreasing function and $|x|_1 \leq B - k$, we have

$$\begin{aligned} Pr(W_x > B) &= \frac{1}{2} \operatorname{erfc}\left(\frac{B - |x|_1}{|x|_1 \cdot \sigma\sqrt{2}}\right) \\ &\leq \frac{1}{2} \operatorname{erfc}\left(\frac{k}{|x|_1 \cdot \sigma\sqrt{2}}\right) \\ &\leq \frac{1}{2} \operatorname{erfc}\left(\frac{k}{n\sigma\sqrt{2}}\right) \end{aligned} \tag{8}$$

Since $r_k = \frac{k}{n\sigma\sqrt{2}} > 0$, we can use the upper bound for the erfc from [25],

$$\begin{aligned} Pr(W_x > B) &\leq \frac{1}{2} \operatorname{erfc}\left(\frac{k}{n\sigma\sqrt{2}}\right) \\ &\leq \frac{1}{\sqrt{\pi}} \frac{e^{-r_k^2}}{r_k + \sqrt{r_k^2 + \frac{4}{\pi}}} \end{aligned}$$

$$\leq \frac{1}{\sqrt{\pi}} e^{\frac{-k^2}{2n^2\sigma^2}}.$$

From Equation 8, we have $Pr(W_x > B \mid |x|_1 = B) = \frac{1}{2} \operatorname{erfc}\left(\frac{B-|x|_1}{|x|_1 \cdot \sigma\sqrt{2}}\right)$ which is $\frac{1}{2}$. □

The following two figures are the experimental results for the constraint value $B = 85$.

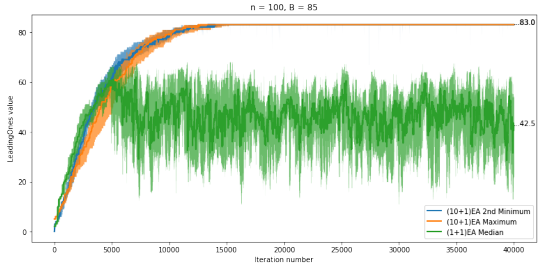

Figure 6: (10+1) EA and (1+1) EA on LeadingOnes with $n = 100$, $B = 85$ and $N(1, 0.1)$ chance constraint for 40000 iterations.

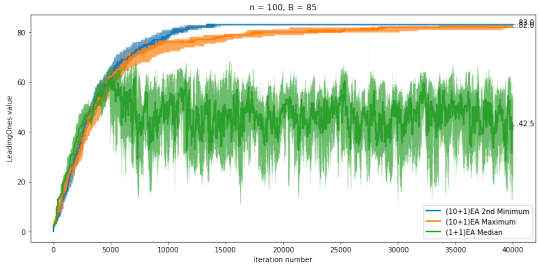

Figure 7: (10+1) EA and (1+1) EA on LeadingOnes with $n = 100$, $B = 85$ and $U(B - \sqrt{3}, B + \sqrt{3})$ stochastic constraint for 40000 iterations.