Runtime Analysis of Genetic Algorithms with Very High Selection Pressure

Anton V. Eremeev[1,2]

[1] Sobolev Institute of Mathematics, Omsk Branch,
13 Pevtsov str., 644099, Omsk, Russia
[2] Omsk State University n.a. F.M. Dostoevsky,
55A Mira Ave., 630077, Omsk, Russia
`eremeev@ofim.oscsbras.ru`

Abstract. The paper is devoted to upper bounds on the expected first hitting time of the sets of optimal solutions for non-elitist genetic algorithms with very high selection pressure. The results of this paper extend the range of situations where the upper bounds on the expected runtime are known for genetic algorithms and apply, in particular, to the Canonical Genetic Algorithm. The obtained bounds do not require the probability of fitness-decreasing mutation to be bounded by a constant less than one.

Keywords: Combinatorial optimization, Evolutionary algorithms, Runtime analysis, Fitness Level.

Introduction

The genetic algorithms (GAs) are randomized heuristic algorithms employing a population of tentative solutions (individuals), which is iteratively updated by means of selection, mutation and crossover operators, thus simulating an evolutionary type of search for optimal or near-optimal solutions. Different modifications of GAs are widely used in areas of operations research and artificial intelligence. A wider class of *evolutionary algorithms* (EAs), has a more flexible outline, possibly neglecting the crossover operator and admitting a population which consists of a single individual. Two major types of evolutionary algorithm outline are now well-known: the *elitist* EAs keep a certain number of "most promising" individuals from the previous iteration, while the *non-elitist* EAs compute all individuals of a new population independently using the same randomized procedure.

The theoretical analysis of GAs has been subject to increasing interest over the last two decades and several different approaches have been developed. A significant progress in understanding of non-elitist GAs was made in [28] by means of dynamical systems. However most of the findings in [28] apply to the infinite population case, so it is not clear how these results can be used to estimate the runtime of GAs, i.e. the expected number of individuals computed and evaluated until the optimum is

---

In: A. Kononov et al. (eds.): DOOR 2016, Vladivostok, Russia, published at http://ceur-ws.org

found for the first time. A theoretical possibility of constructing GAs that provably optimize an objective function with high probability in polynomial time was shown in [27] using rapidly mixing Markov chains. However [27] provides only a very simple artificial example where this approach is applicable. The drift analysis was first adapted to studying elitist EAs in [15] and further extended in [8, 20] to non-elitist EAs without a crossover.

A series of works has attempted to show that the use of crossover operator in GAs and other evolutionary algorithms can reduce their runtime (see e.g. [16], [19], [22], [26]) but most of the positive results apply to families of problem instances with specific structure. At the same time [23] showed that a well-known non-elitist GA with proportional selection operator is inefficient on one of the most simple benchmark functions ONEMAX, even when the crossover is used. Upper bounds obtained in [7] for the runtime of GAs with crossover match (up to a constant factor) the analogous upper bounds known for mutation-only GAs [8]. In [10], sufficient conditions are found under which a non-elitist GA with tournament selection first visits a local optimum of a pseudo-Boolean function in polynomially bounded time on average. The runtime bound from [10] indicates that if a local optimum is efficiently computable by the local search method, it is also computable in expected polynomial time by a GA with tournament selection.

In the present paper, the genetic algorithms are studied on a wide class of combinatorial optimization problems. The expected number of tentative solutions constructed, until first visiting a desired area for the first time, is considered as the main criterion of GA efficiency. Such an area may consist of locally optimal solutions or of globally optimal solutions or of feasible solutions with sufficiently small relative error. The main result is obtained by combining the approaches from [10, 21] and applies to a wider range of selection operators, compared to the result from [10], including the proportional selection of Canonical GA [13] (the term “Canonical GA” was coined in [24]). Considering the selection operators with very high selection pressure, in this paper, we can neglect the probability of downgrading mutations although this probability needs to be taken into account in [7, 8]. By downgrading mutations here we mean mutations which decrease the quality of solutions (a formal definition of downgrading mutation will be given further). In contrast to [7, 8], here we consider explicitly the constrained optimization problems.

In most general setting, a combinatorial optimization problem with maximization criterion is formulated as follows:

$$\max\{F(x) \mid x \in \mathrm{Sol}\}, \tag{1}$$

where $\mathrm{Sol} \subseteq \mathcal{X}$ is the set of feasible solutions, $\mathcal{X} = \{0,1\}^n$ is the search space, $F(\cdot)$ is the objective function. The optimal value of the criterion is denoted by F^*. The minimization problems are formulated analogously. Without loss of generality, by default we will consider the maximization problems. The results will hold for the minimization problems as well.

Genetic Algorithms. In the process of the GA execution, a sequence of populations $P^t = (x^{1,t}, \dots, x^{\lambda,t})$, $t = 0, 1, \dots$, is computed, where each population consists of λ *genotypes.* In the present paper, by the genotypes we mean the elements of the search space $\mathcal{X}$, and *genes* $x_i,\ i \in [n]$ are the components of a genotype $x \in \mathcal{X}$. Here and below, we use the notation $[n] := \{1, 2, ..., n\}$ for any positive integer n.

An initial population P^0 consists of randomly generated genotypes, and every next population is constructed, on the basis of the previous one. For convenience of GA description, in what follows we assume that the population size λ is even.

In each iteration of a GA, $\lambda/2$ pairs of parent genotypes are chosen from the current population P^t using the randomized *selection* procedure $\mathrm{Sel} : \mathcal{X}^\lambda \to [\lambda]$. In this procedure, a parent genotype is independently drawn from the previous population P^t where each individual is assigned a selection probability depending on its *fitness* $f(x)$. Usually a higher fitness value of an individual implies higher (or equal) selection probability. We assume that the fitness function is defined on the basis of objective function. If $x \in \mathrm{Sol}$ then $f(x) = \phi(F(x))$, where $\phi : \mathbb{R} \to \mathbb{R}_+$ is a monotone increasing function in the case of maximization problem or a monotone decreasing function in the case of minimization problem. Otherwise (i.e. if $x \notin \mathrm{Sol}$), the fitness incorporates some penalty, which ensures that $f(x) < \min_{y \in \mathrm{Sol}} F(y)$.

Given the current population, each pair of offspring genotypes is created independently from other pairs using the randomized operators of *crossover* and *mutation*. Some authors consider crossover operators that output a single genotype (see e.g. [7], [4], [17], [23], [28]), while others consider crossovers with two output genotypes (see e.g. [3], [13], [24], [27]). For the sake of uniform treatment of both versions of crossover, let us denote the number of output genotypes by $r,\ r \in \{1, 2\}$. In what follows, we assume that $\mathrm{Cross} : \mathcal{X} \times \mathcal{X} \to \mathcal{X}^r$ and $\mathrm{Mut} : \mathcal{X} \to \mathcal{X}$ are efficiently computable by randomized routines. When a new population P^{t+1} is constructed, the non-elitist GA proceeds to the next iteration.

Algorithm 1. Non-Elitist Genetic Algorithm in the case of $r = 2$

```
Generate the initial population $P^0$, assign $t := 1$.
While a termination condition is not met do:
    Iteration $t$.
    For $j$ from 1 to $\lambda/2$ do:
        Selection: $i := \mathrm{Sel}(P^t)$, $i' := \mathrm{Sel}(P^t)$.
        Crossover: $(x, y) := \mathrm{Cross}(x^{it}, x^{i't})$.
        Mutation: $x^{2j-1,t+1} := \mathrm{Mut}(x)$,  $x^{2j,t+1} := \mathrm{Mut}(y)$.
    End for.
    $t := t + 1$.
End while.
```

Algorithm 2. Non-Elitist Genetic Algorithm in the case of $r = 1$

```
Generate the initial population $P^0$, assign $t := 1$.
While a termination condition is not met do:
    Iteration $t$.
    For $j$ from 1 to $\lambda$ do:
        Selection: $i := \mathrm{Sel}(P^t)$, $i' := \mathrm{Sel}(P^t)$.
        Crossover: $x := \mathrm{Cross}(x^{it}, x^{i't})$.
        Mutation: $x^{j,t+1} := \mathrm{Mut}(x)$.
    End for.
    $t := t + 1$.
End while.
```

The output of a GA is an individual with the maximum fitness value in all populations constructed until the termination condition was met.

In the theoretical analysis of GAs it is often assumed that the algorithm constructs an infinite sequence of populations and the termination condition is never met. In practice, the termination condition is required not only to stop the search and output the result, but also to perform multiple restarts of the GA with random initialization [3, 5]. Multiple independent runs of randomized algorithms or local search (multistart) are widely used to prevent localization of the search in the "unpromising" areas of the search space (see e.g. [6]) and applicability of multistart to the evolutionary algorithms has some theoretical basis [9, 27].

In this paper, together with the standard version of Non-Elitist GA (Algorithms 1 and 2), we study the *GA with multistart*, where a GA outlined as Algorithm 1 or 2 is ran independently from the previous executions for an unlimited number of times. The stopping criterion in Algorithm 1 or 2 in this case is the iterations limit $t \le t_{\max}$, where $t_{\max}$ is a tunable parameter.

In what follows, we consider three options for selection operator: the tournament selection [14], the (μ, λ)-selection [21] and the proportional selection [13]. In *k-tournament selection*, k individuals are sampled uniformly at random with replacement from the population, and the fittest of these individuals is returned. The tunable parameter k is called the *tournament size*. In *(μ, λ)-selection*, parents are sampled uniformly at random among the fittest μ individuals in the population P^t. In the case of proportional selection,

$$\Pr(\mathrm{Sel}(P^t) = i) := \frac{f(x^{it})}{\sum_{j=1}^{\lambda} f(x^{jt})}, \tag{2}$$

if $\sum_{j=1}^{\lambda} f(x^{jt}) > 0$; otherwise the index of the parent individual is chosen uniformly at random.

Canonical Genetic Algorithm proposed in [13] corresponds to the GA outline with $r = 2$, where all individuals of the initial population are chosen independently and uniformly from $\mathcal{X}$. This GA uses the proportional selection, a *single-point crossover* Cross^* and a *bitwise mutation* Mut^*. The last two operators work as follows.

The single-point crossover computes $(x', y') = \text{Cross}^*(x, y)$ for two input genotypes $x = (x_1, ..., x_n)$, $y = (y_1, ..., y_n)$, so that with a given probability p_c,

$$x' = (x_1, ..., x_\chi, y_{\chi+1}, ..., y_n), \quad y' = (y_1, ..., y_\chi, x_{\chi+1}, ..., x_n),$$

where the random position χ is chosen uniformly from 1 to $n-1$. With probability $1 - p_c$ both parent individuals are copied without any changes, i.e. $x' = x,\ y' = y$.

The bitwise mutation Mut^* computes a genotype $x' = \text{Mut}^*(x)$, where independently of other bits, each bit $x'_i,\ i \in [n]$, is assigned a value $1 - x_i$ with probability p_m and with probability $1 - p_m$ it keeps the value x_i. The tunable parameter p_m is also called the *mutation rate*. Choosing the mutation rate, many authors assume $p_m = 1/n$.

Another well-known operator of *point mutation* with a given probability p_m modifies one randomly chosen bit, otherwise (with probability $1-p_m$) the given genotype remains unchanged.

The following condition holds for many well-known crossover operators: there exists a positive constant ε_0 which does not depend on the problem instance, such that the output of crossover $(x', y') = \text{Cross}(x, y)$ satisfies the inequality

$$\varepsilon_0 \le \Pr\Big(\max\{f(x'), f(y')\} \ge \max\{f(x), f(y)\}\Big). \tag{3}$$

for any $x, y \in \mathcal{X}$. Condition (3) suggests that the fitness of at least one of the genotypes resulting from crossover $(x', y') = \text{Cross}(x, y)$ is not less than the fitness of the parents $x, y \in \mathcal{X}$ with probability at least ε_0. This condition is fulfilled for the single-point crossover with $\varepsilon_0 = 1 - p_c$, if $p_c < 1$ is a constant. In the case of crossover operator with a single output genotype $x' = \text{Cross}(x, y)$ analogous condition is as follows

$$\varepsilon_0 \le \Pr\Big(f(x') \ge \max\{f(x), f(y)\}\Big). \tag{4}$$

Condition (4) is also satisfied with $\varepsilon_0 = 1$ for the optimized crossover operators, where at least one of the two offspring is computed as a solution to optimal recombination problem (see e.g., [1, 3, 11]). It was shown in [7] that for some well-known crossover operators and simple fitness functions condition (4) holds with $\varepsilon_0 = 1/2$.

1 The Main Result

This section generalizes the analysis of Non-Elitist Genetic Algorithm carried out in [10], adapting it to different selection operators and making it applicable to the GAs with multistart, which allows us to deal with both feasible and infeasible solutions.

Suppose that for some m there is an ordered partition of $\mathcal{X}$ into subsets $A_0, \ldots, A_{m+1}$ called *levels* [7]. Level A_0 may be an empty set. Level A_{m+1} will be the target level in subsequent analysis. The target level may be chosen as the set of solutions with maximal fitness or the set of local optima or the set of ρ-approximation solutions for some approximation factor $\rho > 1$ (a feasible solution y to a maximization problem is called a ρ-approximation solution if it satisfies the inequality $F^*/F(y) \le \rho$). A well-known example of partition is the *canonical* partition, where $A_0 = \emptyset$ and each level $A_j,\ j \in [m+1]$ regroups solutions having the same fitness value (see e.g. [8, 25]). In what follows, level A_0 may be used to encompass the set of infeasible solutions.

In this paper, we will often use values which are independent of an instance of problem (1) and of a levels partition, but completely determined by the GA outline and its operators. Such values will be called constants. The same applies to the constants in $O(\cdot)$ notation. It will be convenient to use the symbol $H_j := \cup_{i=j}^{m+1} A_i$ for the union of all levels starting from level j, $j \in [m+1]$. The symbol e in what follows denotes the base of the natural logarithm.

Extending the notation from [7, 21], we will define the *selective pressure* $\beta(0,P)$ of a selection operator $\mathrm{Sel}(P)$ as the probability of selecting an individual that belongs to the highest level occupied by the individuals of P.

Theorem 1. *Given a partition* $A_0, \dots, A_{m+1}$ *of* $\mathcal{X}$*, let there exist parameters* s_*, p_1, ε *and* β_0 *from* $(0,1]$*, such that for any* $j \in [m]$*:*

(C1) $\Pr(\mathit{Mut}(x) \in H_{j+1}) \ge s_*$ *for any* $x \in A_j$*,*
(C2) $\Pr\left(x^{i,0} \in H_1 \text{ for some } i \in [\lambda]\right) \ge p_1$*,*
(C3) $\beta(0,P) \ge \beta_0$ *for any* $P \in (\mathcal{X} \backslash A_{m+1})^\lambda$*,*
(C4) $\lambda \ge \frac{2(1+\ln m)}{s_* \varepsilon \beta_0 (2-\beta_0)}$*,*
(C5) *for any* $(x,y) \in (H_j \times \mathcal{X}) \cup (\mathcal{X} \times H_j)$

$$\varepsilon \le \begin{cases} \Pr\Big(\mathit{Cross}(x,y) \in H_j\Big), & \text{in case of} \quad r = 1, \\ \Pr\Big(\mathit{Cross}(x,y) \in (H_j \times \mathcal{X}) \cup (\mathcal{X} \times H_j)\Big), & \text{in case of} \quad r = 2. \end{cases}$$

Then with probability not less than p_1/e *at least one of the populations* $P^0, P^1, \dots, P^m$ *contains an individual from* A_{m+1}*.*

Let us informally describe the conditions of the theorem. Condition (C1) requires that for each level j, there is a lower bound s_* on the "upgrade" probability from level j. Condition (C2) ensures that at least one individual of the initial population is above level 0 with probability not less than p_1. Condition (C3) requires that the selective pressure induced by the selection mechanism is sufficiently high. Condition (C4) requires that the population size λ is sufficiently large. Condition (C5) is a level-based analog of inequalities (3) and (4). This condition follows from (3) or (4) with $\varepsilon = \varepsilon_0$ in the case of the canonical partition.

Proof of Theorem 1. For any $t = 0, 1, \dots$ let the event E_i^{t+1}, $i \in [\lambda/2]$, consist in fulfilment of the following three conditions when the i-th pair of offspring is computed:

1. At least one of the two parents is chosen from the highest level A_j occupied by population P^t.
2. When the crossover operator is applied, at least one of its outputs belongs to H_j. W.l.o.g. we assume that this output is x.
3. The mutation operator applied to x produces a genotype in H_{j+1}.

Let p denote the probability of the union of events $E_i^{t+1},\ i \in [\lambda/2]$. In what follows, we will construct a lower bound $\ell \le p$, which holds for any population P^t. According to the outline of GA, $\Pr(E_1^{t+1}) = \dots = \Pr(E_{\lambda/2}^{t+1})$. Let us denote this probability by q. Note that q is bounded from below by $s_*\varepsilon(1-(1-\beta_0)^2) = s_*\varepsilon\beta_0(2-\beta_0)$. Given a population P^t, the events E_j^{t+1}, $j = 1, \dots, \lambda/2$, are independent, so $p \geq 1-(1-q)^{\lambda/2} \geq 1 - e^{-q\lambda/2}$. In what follows we shall use the fact that condition (C4) implies

$$\lambda \geq \frac{2}{s_*\varepsilon\beta_0(2-\beta_0)} \geq 2/q. \tag{5}$$

To bound probability p from below, we first note that for any $z \in [0,1]$ holds

$$1 - \frac{z}{e} \geq e^{-z}. \tag{6}$$

Assume $z = e^{-q\lambda/2+1}$. Then in view of inequality (5), $z \leq 1$, and consequently,

$$p \geq \exp\left\{-e^{1-q\lambda/2}\right\} \geq \exp\left\{-e^{1-s_*\varepsilon\beta_0(2-\beta_0)\lambda/2}\right\}. \tag{7}$$

We will use the right-hand side expression of (7) as the lower bound ℓ for p.

For any $t = 1, 2, \dots$ let us define the event $\mathcal{E}_t := E_1^t + \dots + E_{\lambda/2}^t$. Note that event $\mathcal{E}_t$ captures some of the possible scenarios of "upgrading" the best individuals of the current population to the next level. Besides that, let $\mathcal{E}_0$ denote the event that $x^{i,0} \in H_1$ for some $i \in [\lambda]$. Then the probability to reach the target level A_{m+1} in a series of at most m iterations is lower bounded by $\Pr(\mathcal{E}_0 \& \dots \& \mathcal{E}_m)$ and

$$\Pr(\mathcal{E}_0 \& \dots \& \mathcal{E}_m) = \Pr(\mathcal{E}_0) \prod_{t=0}^{m-1} \Pr(\mathcal{E}_{t+1} | \mathcal{E}_0 \& \dots \& \mathcal{E}_t) \geq p_1 \ell^m. \tag{8}$$

in view of condition (C2). Now using condition (C4), we get:

$$\ell^m = \exp\left\{-me^{1-s_*\varepsilon\beta_0(2-\beta_0)\lambda/2}\right\} \geq \exp\left\{-me^{-\ln m}\right\} = 1/e.$$

□

An event of *downgrading mutation* of a genotype x may be defined in terms of levels partition as $\mathrm{Mut}(x) \notin H_j$, where j is the level the individual x belongs to. Unlike the results from [7, 8], Theorem 1 is applicable to the GAs where the probability of non-downgrading mutation may tend to zero as the problem size grows. Examples of such operators may be found in highly competitive GAs for Maximum Independent Set Problem and Set Covering Problem [1, 4] and many other GAs in the literature on operations research. Note that in case $|A_{m+1}| = 1$, given an optimal genotype $x \in A_{m+1}$, the bitwise mutation with a constant mutation rate (as used in [1]) causes non-downgrading mutations only with probability $(1-p_{\mathrm{m}})^n = o(1)$ and the mutation operator that inverts m_f bits, where $m_f > 0$ is a given parameter [4], has zero probability of non-downgrading mutations.

2 Lower Bounds for Selection Pressure

The following two propositions may be applied to check condition (C3) in Theorem 1.

Proposition 1. *Let levels* $A_1, \dots, A_m$ *satisfy the monotonicity condition*

$$f(x) < f(y) \text{ for any } x \in A_{j-1},\ y \in A_j,\ j = 2, \dots, m. \tag{9}$$

Then

(i) k-tournament selection with $k \geq \alpha\lambda$*, where the constant* $\alpha > 0$*, satisfies condition (C3) with* $\beta_0 = 1 - e^{-\alpha}$*.*

(ii) (μ, λ)*-selection with a constant parameter* $\mu \leq \lambda$ *satisfies condition (C3) where* $\beta_0 = 1/\mu$*.*

Proof. In the case of k-tournament selection $\beta(0, P) \geq 1 - (1 - 1/\lambda)^k$ and $(1 - 1/\lambda)^k \leq (1 - 1/\lambda)^{\alpha\lambda} \leq e^{-\alpha}$, so part (i) follows. Part (ii) follows from the definition of (μ, λ)-selection immediately. □

The operator of proportionate selection does not have a tunable parameter that allows to set its selection pressure. However such a parameter (let it be ν) may be introduced into the fitness function by assuming that $f(x) = F(x)^\nu$ for any $x \in \text{Sol}$. The proof of the following proposition is similar to that of Lemma 8 in [21]. Here and below $\mathbb{Z}_+$ denotes the set of non-negative integers.

Proposition 2. *Let the levels* $A_1, \dots, A_m$ *satisfy the monotonicity condition (9),* $F : \text{Sol} \to \mathbb{Z}_+$ *and the fitness function is of the form* $f(x) = F(x)^\nu$*, where* $\nu > \max(0, \ln(\alpha\lambda)F^*)$ *for some* $\alpha > 0$*. Then the proportional selection satisfies condition (C3) with* $\beta_0 = 1/(1 + \alpha^{-1})$*.*

Proof. Let F_0^ν be the maximal fitness value in population P and let k denote the number of individuals in P with fitness F_0^ν. The probability to choose one of the fittest individuals is lower bounded as follows

$$\beta(0, P) \geq \frac{kF_0^\nu}{(\lambda - k)(F_0 - 1)^\nu + kF_0^\nu} \geq \frac{k}{\lambda(1 - 1/F_0)^\nu + k} \geq \frac{k}{1/\alpha + k} \geq \frac{1}{1/\alpha + 1},$$

since $(1 - 1/F_0)^\nu \leq (1 - 1/F^*)^\nu \leq e^{-\nu/F^*} \leq 1/(\alpha\lambda)$. □

3 Upper Bounds on Expected Hitting Time of Target Subset

Let T denote the random variable, equal to the number of individuals evaluated until some element of the GA population is sampled from A_{m+1} for the first time.

Corollary 1. *Suppose that conditions (C1)-(C5) of Theorem 1 hold and* $A_0 = \emptyset$*. Then for the GA we have* $E[T] \leq em\lambda$*.*

Proof. Consider a sequence of series of the GA iterations, where the length of each series is m iterations. Suppose, $D_i,\ i = 1, 2, \dots,$ denotes an event of absence of solutions from A_{m+1} in the population throughout the i-th series. The probability of each event $D_i,\ i = 1, 2, \dots,$ is at most $1 - 1/e$ according to Theorem 1. Analogously to bound (8) we obtain the inequality $\Pr(D_1 \& \dots \& D_i) \leq (1 - 1/e)^i$.

Let Y denote the random variable, equal to the number of the first run when a solution form A_{m+1} was obtained. By the properties of expectation (see e.g. [12]),

$$E[Y] = \sum_{i=0}^{\infty} \Pr(Y > i) = 1 + \sum_{i=1}^{\infty} \Pr(D_1 \& \ldots \& D_i) \le 1 + \sum_{i=1}^{\infty} (1 - 1/e)^i = e.$$

Consequently, the average number of iterations until an element of the target subset is first obtained is at most em. □

Assuming $\lambda = \left\lceil \frac{2(1+\ln(m))}{s_* \varepsilon \beta_0 (2-\beta_0)} \right\rceil$ and constant β_0 and ε, Corollary 1 implies $E[T] \le cm \ln(m)/s_*$, where $c > 0$ is a constant. In the special case where $r = 1$ and the probability of non-downgrading mutation $\Pr(\mathrm{Mut}(x) \in H_j \mid x \in A_j),\ j \in [m]$ is lower bounded by a positive constant, the result from [7] gives an upper bound $\mathrm{E}[T] \le c'm \left(\ln(m/s_*) \ln\ln(m/s_*) + 1/s_*\right)$ with some positive constant c'. The latter bound is less demanding to selection pressure and it is asymptotically tighter than the bound $E[T] \le cm \ln(m)/s_*$ e.g. when $s_* \le 1/m$.

Note that the assumption $A_0 = \emptyset$ in Corollary 1 can not be dismissed. Indeed, suppose that $A_0 \ne \emptyset$, and consider a GA where the mutation operator has the following properties. On one hand, it never outputs an offspring in H_1, given an input from A_0. On the other hand, given a genotype $x \in H_1$, the result of mutation is in A_0 with a probability at least c, where $c > 0$. Finally assume that the initialization procedure produces no genotypes from A_{m+1} in population P^0 and the crossover makes no changes to the parent genotypes. Now all conditions of Corollary 1 can be satisfied but with a positive probability of at least c^λ the whole population P^1 consists of solutions from A_0, and subject to this event all populations $P^1, P^2, \ldots$ contain no solutions from H_1. Therefore, $E[T]$ is unbounded.

As an example of usage of Corollary 1 we consider the GA with tournament selection applied to the family of unconstrained optimization problems with objective function LeadingOnes, which is frequently used in the analysis of evolutionary algorithms. The objective function LeadingOnes : $\{0,1\}^n \to \mathbb{Z}_+$ is defined as

$$\textsc{LeadingOnes}(x) = \sum_{i=1}^{n} \prod_{j=1}^{i} x_j$$

i.e. the optimal solution is $x^* = (1, \ldots, 1)$.

Let us use the canonical levels partition: $A_j = \{x \mid F(x) = j-1\}, j \in [n+1], m = n$. Assume that the bitwise mutation operator has the mutation rate $p_{\mathrm{m}} = 1/n$. To move from level A_j to level A_{j+1} under mutation it suffices to modify the first zero bit and not to modify the rest of the bits. So we can use $s_* = (1/n)(1-1/n)^{n-1} = \Omega(1/n)$. Suppose that in the single-point crossover $p_{\mathrm{c}} = 1$. Then in the case of LeadingOnes as it was shown in [7], the constant $\epsilon = 1/2$ satisfies condition (C5). Assuming the tournament size $k = \Theta(\lambda)$, Proposition 1 ensures satisfaction of condition (C3) with a positive constant β_0.

Application of Corollary 1 to the GA with $r = 2$ and $\lambda = \Theta(n \ln(n))$, satisfying (C4), gives the upper bound $E[T] = O(n^2 \ln(n))$. Note that a similar bound obtained in [7] for the case of single-offspring crossover gives $E[T] = O(n^2)$. We expect that extension of the drift analysis [7] to the case of two-offspring crossovers may tighten the upper bound for the LeadingOnes in the case of $r = 2$ as well.

Analogously to Corollary 1 we obtain

Corollary 2. *Let the GA with multistart use the termination condition with* $t_{\max} = m$. *Then* $E[T] \leq em\lambda/p_1$ *holds under conditions (C1)-(C5).*

As an illustrative example for Corollary 2 we consider Canonical GA on the family of instances of Set Cover Problem proposed by E.Balas in [2]. In general the set cover problem (SCP) is formulated as follows. Given: $M = \{1, ..., m\}$ and a set of subsets $M_j \subseteq M$, $j \in [n]$. A subset $J \subseteq [n]$ is called a *cover* if $\cup_{j \in J} M_j = M$. The goal is to find a cover of minimum cardinality. In what follows we denote by N_i the set of indices of the subsets that cover an element i, i. e. $N_i = \{j : i \in M_j\}$ for any i.

In the family $\mathcal{B}(n, p)$ of SCPs introduced by E. Balas in [2], it is assumed that $m = C_n^{p-1}$ and the set $\{N_1, N_2, ..., N_m\}$ consists of all $(n - p + 1)$-element subsets of $[n]$. Thus $J \subseteq [n]$ is an optimal cover iff $|J| = p$.

Family $\mathcal{B}(n, p)$ is known to have a large fractional cover [2] which implies that these SCPs are likely to be hard for integer programming methods. In particular, it was shown in [29] that problems from this class are hard to solve using the L-class enumeration method [18]. When n is even and $p = n/2$, the L-class enumeration method needs an exponential number of iterations in n. In what follows, we analyze GA in this special case.

In the binary encoding of solutions we assume that each bit $x_j \in \{0, 1\}, j \in [n]$, indicates whether j belongs to the encoded set, $J(x) := \{j \in [n] : x_j = 1\}$. If $J(x)$ is a cover then we assume $F(x) = n - |J(x)| + 1$, otherwise we put $F(x) = 0$ as a penalty.

Consider Canonical GA with multistart and scaled fitness function $f(x) = F(x)^\nu$, the termination condition where $t_{\max} = n/2$, a constant parameter $p_{\mathrm{c}} < 1$ and the mutation rate $p_{\mathrm{m}} = 1/n$.

Assume that A_0 is the set of all infeasible solutions and the rest of the levels $A_1, \ldots, A_{m+1}$ are defined according to the canonical partition on Sol, where $m = n/2$. In the case of $p = n/2$, with probability 1/2 a random individual of P^0 is feasible and there exists a constant $p_1 > 0$ satisfying condition (C2). The constant $\epsilon = 1 - p_{\mathrm{c}}$ satisfies condition (C5). The probability that under mutation a genotype from level A_j produces an element of $H_{j+1},\ j \in [m]$ in the case of problems of family $\mathcal{B}(n, p)$ is lower bounded by $s_* = \Omega(1)$. Choosing $\nu > \ln(\alpha\lambda)n/2$ with constant $\alpha > 0$ we ensure condition (C3) according to Proposition 2. Finally, appropriate $\lambda = \Theta(\ln(n))$, satisfies condition (C4). Therefore Corollary 2 implies that an optimal solution is attained for the first time after $\mathrm{E}[T] = O\left(n \ln(n)\right)$ tentative solutions in expectation.

4 Conclusions

The obtained bounds on the first hitting times for the set of optimal solutions are extending some previously known bounds of such kind for genetic algorithms and may be applied to standard benchmarks and genetic algorithms as well as some stat-of-the-art genetic algorithms for combinatorial optimization problems. Considering the selection operators with very high selection pressure, we obtain the bounds that apply even in the cases where the probability of non-downgrading is not lower-bounded by a positive constant.

The paper does not take into account the possible *improvement* of parent solutions in the crossover operator. The further research might address the ways of using this potential.

Acknowledgement

The research is supported by RFBR grant 15-01-00785. The author thanks the participants of the Theory of Evolutionary Algorithms seminar 15211 at Schloss Dagstuhl for helpful comments.

References

1. Aggarwal C.C., Orlin J.B. and Tai R.P.: An optimized crossover for maximum independent set. Operations Research. 45, 225-234 (1997)
2. Balas E.: A sharp bound on the ratio between optimal integer and fractional covers. Mathematics of Operations Research. 9(1), 1-5 (1984)
3. Balas E. and Niehaus W.: Optimized crossover-based genetic algorithms for the maximum cardinality and maximum weight clique problems. Journal of Heuristics. 4(2), 107-122 (1998)
4. Beasley J.E. and Chu P.C.: A genetic algorithm for the set covering problem. European Journal of Operational Research. 94(2), 394-404 (1996)
5. Borisovsky P., Eremeev A., Grinkevich E., Klokov S., and Kosarev N.: Trading hubs construction in electricity markets using evolutionary algorithms. Pattern Recognition and Image Analysis. 24(2), 270-282 (2014)
6. Brimberg J., Hansen P., Mladenovic N.: Attraction probabilities in variable neighborhood search. 4OR. 8(2), 181-194 (2010)
7. Corus D., Dang D.-C., Eremeev A.V., and Lehre P.K.: Level-based analysis of genetic algorithms and other search processes. In: Proc. of Parallel Problem Solving from Nature (PPSN XIII). LNCS Vol. 8672. pp. 912-921. Springer (2014)
8. Dang D.-C. and Lehre P.K., Refined upper bounds on the expected runtime of non-elitist populations from fitness-levels. In: Proc. of Conference on Genetic and Evolutionary Computation (GECCO'14). pp. 1367-1374. ACM, New York, NY (2014)
9. Doerr C. and Lengler J.: Elitist black-box models: analyzing the impact of elitist selection on the performance of evolutionary algorithms. In: Proc. of Genetic and Evolutionary Computation Conference (GECCO). pp. 839-846. ACM, New York (2015)
10. Eremeev A.V.: A genetic algorithm with tournament selection as a local search method (In Russian). In: Proc. of 15-th Baikal International School-Seminar “Optimization methods and their applications”. Vol. 4. pp. 94-99. IDSTU, Irkutsk (2011)
11. Eremeev A.V. and Kovalenko J.V.: Optimal recombination in genetic algorithms for combinatorial optimization problems Part I. Yugoslav Journal of Operations Research. 24(1), 1-20 (2014)
12. Gnedenko B.V.: Theory of probability. Amsterdam, Gordon and Breach, Amsterdam (1997)

13. Goldberg D.E.: Genetic algorithms in search, optimization and machine learning. Addison-Wesley, Reading, MA (1989)
14. Goldberg D.E.: A note on Boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing. Complex Systems. 4, 445-460 (1990)
15. He J. and Yao X.: Drift analysis and average time complexity of evolutionary algorithms. Artificial Intelligence. 127(1), 57-85 2001
16. Jansen T., Wegener I.: On the analysis of evolutionary algorithms – a proof that crossover really can help. Algorithmica. 34(1), 47-66 2002
17. Kochetov Yu.A. and Plyasunov A.V.: Genetic local search the graph partitioning problem under cardinality constraints. Computational Mathematics and Mathematical Physics. 52 (1) 157-167 (2012)
18. Kolokolov A.A.: Regular partitions and cuts in integer programming. In: Discrete Analysis and Operations Research. pp. 59-79. Springer, Netherlands (1996)
19. Kötzing T., Sudholt D., and Theile M.: How crossover helps in pseudo-boolean optimization. In Proc. of the 13th Conference on Genetic and Evolutionary Computation (GECCO-11). pp. 989996. ACM, New York, NY (2011)
20. Lehre, P.K.: Negative drift in populations. In Proc. of the 11th international conference on Parallel problem solving from nature (PPSN-2010). pp. 244253. Springer-Verlag, Berlin, Heidelberg (2010)
21. Lehre P.K.: Fitness-levels for non-elitist populations. In: Proc. of 13th Conference on Genetic and Evolutionary Computation (GECCO-11). pp. 2075-2082 ACM, New York, NY (2011)
22. Lehre P.K. and Yao X.: Crossover can be constructive when computing unique input-output sequences. Soft Computing. 15(9), 16751687 (2011)
23. Oliveto P.S. and Witt C.: On the runtime analysis of the simple genetic algorithm. Theoretical Computer Science. 545, 2-19 (2014)
24. Rudolph G.: Convergence analysis of canonical genetic algorithms. IEEE Transactions on Neural Networks. 5(1), 96-101 (1994)
25. Sudholt D.: A new method for lower bounds on the running time of evolutionary algorithms. IEEE Transactions on Evolutionary Computation. 17(3), 418-435 (2012)
26. Sudholt D.: Crossover speeds up building-block assembly. In Proc. of Genetic and Evolutionary Computation Conference (GECCO-2012). pp. 689702. ACM, New York, NY (2012)
27. Vitányi P.M.B.: A discipline of evolutionary programming. Theoretical Computer Science. 241(1-2), 3-23 (2000)
28. Vose M.D.: The simple genetic algorithm: Foundations and theory. MIT Press, Cambridge, MA (1999)
29. Zaozerskaya L.A.: Investigation and solving of some classes of integer programming problems on the basis of the regular partitions (in Russian). PhD. thesis. Omsk Branch of Sobolev Institute of Mathematics, SB RAS, Omsk (1998)