On Performance Estimates for Two Evolutionary Algorithms

Pavel A. Borisovsky[1] and Anton V. Eremeev[2]

[1] Omsk State University, Mathematical Department,
55 Mira str. 644077, Omsk, Russia
borisovski@mail.ru

[2] Omsk Branch of Sobolev Institute of Mathematics,
13 Pevtsov str. 644099, Omsk, Russia.
eremeev@iitam.omsk.net.ru
http://iitam.omsk.net.ru/~eremeev

Abstract. In this paper we consider the upper and lower bounds on probability to generate the solutions of sufficient quality using evolutionary strategies of two kinds: the (1+1)-ES and the (1,λ)-ES (see e.g. [1, 2]). The results are obtained in terms of monotone bounds [3] on transition probabilities of the mutation operator. Particular attention is given to the computational complexity of mutation procedure for the NP-hard combinatorial optimization problems.

1 Introduction

The estimates for the behaviour of evolutionary strategies (ES) obtained in this paper are based on the a priori known parameters of the mutation operator, called monotone transition bounds. Using a model analogous to the one proposed in [3] we obtain in Sect. 2 the upper and lower bounds on the probability to generate a solution with fitness above given threshold on any iteration of the (1,λ)-ES and the (1+1)-ES. In this section some conditions are formulated when the latter heuristic has definite advantages over the first one. A connection with the complexity theory in Sect. 3 demonstrates some possible limitations on the a priori transition bounds of mutation operators implementable as polynomial-time randomized algorithms [4].

Let the optimization problem consist in finding a feasible solution $y \in Sol \subseteq D$, which maximizes the objective function $f : D \to R$, where D is the *space of solutions*, and $Sol \subseteq D$ is a *set of feasible solutions*. In general an evolutionary algorithm is searching for the optimal or near-optimal solutions using a population of individuals, which is driven by the principles observed in biological evolution. In both simplified versions of ES considered here the population consists of a single individual: in each iteration a new individual is constructed on the basis of the current one, and then the current individual is called a *parent*.

We assume that an individual is represented by a *genotype*, which is a fixed length string g of genes $g^1, g^2, \dots, g^n$, and all genes are the symbols of some finite alphabet A. For example, the alphabet $\Sigma = \{0, 1\}$ is used in many applications.

Each genotype g represents an element $y = y(g)$ of space D, which may not necessarily be a feasible solution.

The search process is guided by evaluations of nonnegative fitness function $\Phi(g)$, which defines the fitness of an individual with genotype g. Usually in case $y(g) \in Sol$, Φ is a monotone function of $f(y(g))$, and for simplicity in this paper we shall assume that $\Phi(g) = f(y(g))$. In case $y(g) \notin Sol$, the fitness function may incorporate a penalty for violation of constraints defining the set Sol. In what follows we will assume that for any $g \in Sol$ and $g' \in D \backslash Sol$, $\Phi(g) > \Phi(g')$ if $D \backslash Sol \neq \emptyset$.

The genotype of the current individual on iteration t of the (1,λ)-ES will be denoted by $b^{(t)}$, and in the (1+1)-ES it will be denoted by $g^{(t)}$. The initial genotypes $b^{(0)}$ and $g^{(0)}$ are generated with some a priori chosen probability distribution. The stopping criterion of the heuristics is the limit of the maximum number of iterations.

The only difference between the (1,λ)-ES and the (1+1)-ES consists in the method of construction of an individual for iteration $t + 1$ using the current individual of iteration t as a parent. In both algorithms the new individual is built with the help of a random *mutation operator* $Mut : A^n \to A^n$, which adds some random changes to the parent genotype. In case of the (1,λ)-ES the mutation operator is independently applied λ times to the parent genotype $b^{(t)}$ and out of λ offspring a single genotype with the highest fitness value is chosen as $b^{(t+1)}$. In case there are several offspring with the highest fitness, the new individual $b^{(t+1)}$ is chosen with uniform distribution among them. In the (1+1)-ES heuristic the mutation operator is applied to $g^{(t)}$ once. If $g = Mut(g^{(t)})$ is such that $\Phi(g) > \Phi(g^{(t)})$, then we set $g^{(t+1)} := g$; otherwise $g^{(t+1)} := g^{(t)}$.

In this paper the mutation operator will be viewed as a randomized algorithm which has an input g and a random output $Mut(g) \in A^n$ with probability distribution depending on g. We assume that all data of the given problem instance is available in mutation as well. One of the frequently used mutation operators consists in randomly changing each gene of g with a fixed mutation probability p_m. Another example is the operator which chooses a random position i and replaces the gene g^i by a new random symbol (see e.g. [5]).

The analysis of ES in principle could be carried out by the means of Markov chains theory (see e.g. [5]). However, the size of the transition matrix of a Markov chain grows exponentially as the genotype length increases, and the applicability of this approach appears to be limited when studying the optimization problems with large cardinality of solutions space. In order to overcome this difficulty we use a grouping of the states into larger classes on the basis of fitness.

1.1 Notations and Assumptions of the Model

The model of mutation operator used in this paper has been introduced in [3]. The information about fitness of distinct genotypes is not used explicitly. Instead, model makes use of certain a priori known parameters of probability distribution of the mutation operator described below.

Assume that there are d level lines of the fitness function fixed so that $\Phi_0 = 0 < \Phi_1 < \Phi_2 \ldots < \Phi_d$. The number of the level lines and the fitness values corresponding to them may be chosen arbitrarily, but they should be relevant to the given problem and the mutation operator to yield a meaningful model. Let us introduce the following sequence of subsets of the set A^n:

$$H_i = \{g : \Phi(g) \geq \Phi_i\}, \quad i = 0, \ldots, d.$$

Due to nonnegativity of the fitness function, H_0 equals to the set of all genotypes. Besides that, for the sake of convenience, let us define the set $H_{d+1} = \emptyset$.

Now suppose that for all $i = 0, ..., d$ and $j = 1, ..., d$ the a priori lower bounds α_{ij} and upper bounds β_{ij} on mutation transition probability from subset $H_i \backslash H_{i+1}$ to H_j are known, i.e. for every $g \in H_i \backslash H_{i+1}$ holds $\alpha_{ij} \leq P\{Mut(g) \in H_j\} \leq \beta_{ij}$, where $P\{Mut(g) \in H_j\} = \sum_{g' \in H_j} P\{Mut(g) = g'\}$.

Let $\mathbf{A}$ denote the matrix with elements α_{ij} where $i = 0, ..., d$, and $j = 1, ..., d$. The matrix of upper bounds β_{ij} is denoted by $\mathbf{B}$.

If for all $i = 0, \ldots, d$, and $j = 1, \ldots, d$ the probability $P\{Mut(g) \in H_j\}$ does not depend on choice of $g \in H_i \backslash H_{i+1}$ then there exist such matrices of probability bounds that $\mathbf{A} = \mathbf{B}$. In this case the mutation operator is called a *step mutation operator with respect to the sequence of subsets* $H_0, H_1, \ldots, H_d$ (or step mutation operator for short), and the matrix $\mathbf{A} = \mathbf{B}$ is called *the threshold transition matrix*. Here we should note that the step mutation operators do not seem to be natural for the real-life optimization problems, however they will be useful in further analysis as a sort of ideal operators.

A matrix $\mathbf{M}$ with elements $m_{ij}, i = 0, \ldots, d$, and $j = 1, \ldots, d$ will be called *monotone* if $m_{i-1,j} \leq m_{ij}$ for all i, j from 1 to d. In other words, the matrix of bounds on transition probabilities is monotone if for any $j = 1, \ldots, d$, the genotypes from any subset H_i have the bounds on transition probabilities to H_j not less than the bounds of the genotypes from any $H_{i'}$, $i' < i$.

In what follows we will focus the attention on the monotone matrices. Obviously, for any mutation operator the monotone bounds exist. (For example $\mathbf{A} = \mathbf{0}$ where $\mathbf{0}$ is a zero matrix and $\mathbf{B} = \mathbf{U}$ where $\mathbf{U}$ is the matrix with all elements equal 1.) The problem may be only with the absence of bounds which are sharp enough to evaluate the mutation operator properly. In fact if we conjecture that the probability distribution of a mutation operator is closely approximated by some monotone matrix of bounds on transition probabilities, this will imply that all solutions with high fitness values are relatively "easy" to reach from other solutions of high fitness. So the assumption that such "good" approximation exists, appears to be resembling the well-known "big valley" conjecture.

The distribution of the current individual in the (1,λ)-ES will be characterized (though not completely) by the *vector of probabilities*

$$P^{(t)} = (p_1^{(t)}, ..., p_d^{(t)}) = (P\{b^{(t)} \in H_1\}, \ldots, P\{b^{(t)} \in H_d\}),$$

which reflects the chances to have "good enough" genotypes on iteration t. By "good" genotypes we mean the genotypes with fitness above a certain threshold.

In [3] for a genetic algorithm (GA) based on the tournament selection and mutation it was shown that in the case of step mutation with monotone threshold transition matrix and a sufficiently large population, a GA with a smaller tournament size does not "outperform" the same GA with larger tournament size. Namely, for the first algorithm the probability that the offspring generated on iteration t belongs to $H_j, j = 1, ..., d$ is not greater than that for the latter one. This conclusion motivated our present study of the (1,λ)-ES, since its behavior is identical to the behavior of the GA with infinite tournament size in the framework of our model.

2 Bounds for the Probability of Obtaining "Good" Solutions

Our goal now is to estimate $P\{b^{(t)} \in H_j\}$ for all j. Note that there always exists such $0 \le i \le d$ that $b^{(t-1)} \in H_i \setminus H_{i+1}$, so by the total probability formula

$$p_j^{(t)} = \sum_{i=0}^{d} P\{b^{(t)} \in H_j | b^{(t-1)} \in H_i \setminus H_{i+1}\} P\{b^{(t-1)} \in H_i \setminus H_{i+1}\}.$$

Let us assume for convenience that $p_0^{(t-1)} = 1$ and $p_{d+1}^{(t-1)} = 0$. Then for all $i = 0, ..., d$ we have $P\{b^{(t-1)} \in H_i \setminus H_{i+1}\} = p_i^{(t-1)} - p_{i+1}^{(t-1)}$. Thus

$$P\{b^{(t)} \notin H_j | b^{(t-1)} \in H_i \setminus H_{i+1}\} = \sum_{b \in H_i \setminus H_{i+1}} \frac{P\{b^{(t)} \notin H_j | b^{(t-1)} = b\} P\{b^{(t-1)} = b\}}{P\{b^{(t-1)} \in H_i \setminus H_{i+1}\}} \le$$

$$\sum_{b \in H_i \setminus H_{i+1}} \frac{(1-\alpha_{ij})^\lambda P\{b^{(t-1)} = b\}}{P\{b^{(t-1)} \in H_i \setminus H_{i+1}\}} = (1-\alpha_{ij})^\lambda.$$

Proposition 1. *The components of $P^{(t)}$ are bounded as follows:*

$$p_j^{(t)} \ge 1 - (1-\alpha_{0j})^\lambda + \sum_{i=1}^{d} ((1-\alpha_{i-1,j})^\lambda - (1-\alpha_{ij})^\lambda) p_i^{(t-1)} \qquad (1)$$

for $j = 1, \ldots, d$, and (1) is an equality in case of a step mutation operator.

Let us define for $(d \times d)$-matrix $\mathbf{W}$ a matrix norm $||\mathbf{W}|| = \max_j \sum_{i=1}^{d} |w_{ij}|$. To convert recurrent bound (1) into a bound for a general term of sequence $\{P^{(t)}\}$ we will use the following result from [3].

Lemma 1. *Let $\mathbf{W}$ be a $(d \times d)$-matrix with nonnegative elements such that $||\mathbf{W}|| < 1$. Suppose that $\alpha \in R^d$, $\beta \in R^d$ and $\{\zeta^{(t)}\}, t = 0, 1, \ldots$ is a sequence of vectors from R^d.*
(a) If $\zeta^{(t+1)} \ge \zeta^{(t)}\mathbf{W} + \alpha$ for all $t \ge 0$, then

$$\zeta^{(t)} \ge \zeta^{(0)}\mathbf{W}^t + \alpha(\mathbf{I} - \mathbf{W})^{-1}(\mathbf{I} - \mathbf{W}^t). \qquad (2)$$

(b) If $\zeta^{(t+1)} \leq \zeta^{(t)}\mathbf{W} + \beta$ *for all* $t \geq 0$*, then*

$$\zeta^{(t)} \leq \zeta^{(0)}\mathbf{W}^t + \beta(\mathbf{I}-\mathbf{W})^{-1}(\mathbf{I}-\mathbf{W}^t), \tag{3}$$

were $\mathbf{I}$ *denotes the identity matrix.*

Note that the right-hand side of (2) approaches $\alpha(\mathbf{I}-\mathbf{W})^{-1}$ when t tends to infinity, regardless of the initial vector $\zeta^{(0)}$.

Now we apply Lemma 1 to matrix $\mathbf{W}'$ with elements $w'_{ij} = (1 - \alpha_{i-1,j})^\lambda - (1-\alpha_{i,j})^\lambda$ (which are nonnegative if $\mathbf{A}$ is monotone), and $\alpha' = (1-(1-\alpha_{01})^\lambda, \ldots, 1-(1-\alpha_{0d})^\lambda)$. Also let us require that $\alpha_{dj} - \alpha_{0j} < 1, j = 1, \ldots, d$ which is sufficient for $||\mathbf{W}|| < 1$.

Theorem 1. *If* $\mathbf{A}$ *is monotone and* $\alpha_{dj} - \alpha_{0j} < 1$ *for* $j = 1, \ldots, d$*, then* $P^{(t)} \geq \zeta^{(t)}$ *where* $\zeta^{(t)}$ *is bounded according to* (2) *with* $\zeta^{(0)} = P^{(0)}, \mathbf{W} = \mathbf{W}'$*, and* $\alpha = \alpha'$*.*

Note that in many practical algorithms an arbitrary given genotype may be produced with non-zero probability during mutation. In such cases the condition $\alpha_{dj} - \alpha_{0j} < 1$ is obviously satisfied for all j.

The upper bounds for the components of $P^{(t)}$ can be obtained from matrix $\mathbf{B}$ in the same way. Let us denote $\hat{w}'_{ij} = (1-\beta_{i-1,j})^\lambda - (1-\beta_{i,j})^\lambda$, $i = 1, ..., d$, $j = 1, ..., d$, and $\beta' = (1-(1-\beta_{01})^\lambda, \ldots, 1-(1-\beta_{0d})^\lambda)$.

Theorem 2. *If* $\mathbf{B}$ *is monotone and* $\beta_{dj} - \beta_{0j} < 1$ *for* $j = 1, \ldots, d$*, then* $P^{(t)} \leq \zeta^{(t)}$ *where* $\zeta^{(t)}$ *is bounded according to* (3) *with* $\zeta^{(0)} = P^{(0)}, \mathbf{W} = \widehat{\mathbf{W}}'$*, and* $\beta = \beta'$*.*

In case of step mutation operator we have $\mathbf{A} = \mathbf{B}$, and the lower and upper bounds of theorems 1 and 2 coincide. In this case we can consider $P^{(t)}$ as a function on $P^{(0)}$, and denote $P^{(t)} = P^{(t)}(P^{(0)})$.

One can note that in the case of step monotone mutation on any iteration t there is no use to continue generating the offspring genotypes if we have already found a genotype with higher fitness than $\Phi(b^{(t)})$ (then it is reasonable to use the new genotype as a parent). Besides that if all offspring have lower fitness than $\Phi(b^{(t)})$, there is no sense to substitute $b^{(t)}$ by the best offspring.

This observation leads us to investigation of the (1+1)-ES and to its comparison to the (1,λ)-ES in a formal setting. For the (1+1)-ES we define the following vector of probabilities:

$$Q^{(t)} = (q_1^{(t)}, ..., q_d^{(t)}) = (P\{g^{(t)} \in H_1\}, \ldots, P\{g^{(t)} \in H_d\}).$$

By means of the total probability formula we conclude that

$$q_j^{(t)} \geq \sum_{i=0}^{j-1} \alpha_{ij} P\{g^{(t)} \in H_i \setminus H_{i+1}\} + q_j^{(t-1)} = \sum_{i=0}^{j-1} \alpha_{ij}(q_i^{(t-1)} - q_{i+1}^{(t-1)}) + q_j^{(t-1)}.$$

Proposition 2. *The components of $Q^{(t)}$ are bounded as follows:*

$$q_j^{(t)} \geq \alpha_{0j} + \sum_{i=1}^{j-1} (\alpha_{ij} - \alpha_{i-1,j}) q_i^{(t-1)} + (1 - \alpha_{j-1,j}) q_j^{(t-1)} \tag{4}$$

for $j = 1, \ldots, d$, and (4) is an equality in case of step mutation operator.

We can also apply Lemma 1 defining matrix $\mathbf{W}''$ and vector α'' as follows:

$$w''_{ij} = \begin{cases} \alpha_{i,j} - \alpha_{i-1,j}, & i < j, \\ 1 - \alpha_{j-1,j}, & i = j, \\ 0, & i > j, \end{cases} \qquad \alpha'' = (\alpha_{01}, \ldots, \alpha_{0d}).$$

Theorem 3. *If $\mathbf{A}$ is monotone and $\alpha_{0j} > 0$ for $j = 1, \ldots, d$, then $Q^{(t)} \geq \zeta^{(t)}$ where $\zeta^{(t)}$ is bounded according to (2) with $\zeta^{(0)} = Q^{(0)}, \mathbf{W} = \mathbf{W}''$, and $\alpha = \alpha''$.*

It is easy to verify that $\mathbf{1} = \alpha'' + \mathbf{1}\mathbf{W}''$, where $\mathbf{1} = (1, \ldots, 1)$. Thus, naturally, $P\{g^{(t)} \in H_d\} \to 1$ when $t \to \infty$.

Analogously let us denote

$$\hat{w}''_{ij} = \begin{cases} \beta_{i,j} - \beta_{i-1,j}, & i < j, \\ 1 - \beta_{j-1,j}, & i = j, \\ 0, & i > j, \end{cases} \qquad \beta'' = (\beta_{01}, \ldots, \beta_{0d}).$$

Theorem 4. *If $\mathbf{B}$ is monotone and $\beta_{0j} > 0$ for $j = 1, \ldots, d$, then $Q^{(t)} \leq \zeta^{(t)}$ where $\zeta^{(t)}$ is bounded according to (3) with $\zeta^{(0)} = Q^{(0)}, \mathbf{W} = \widehat{\mathbf{W}}''$, and $\beta = \beta''$.*

2.1 Comparison of the (1+1)-ES with the (1,λ)-ES

The superiority of the (1+1)-ES over the (1,λ)-ES in case of monotone step mutation has been mentioned above and now it will be formally supported. To compare the performance of both algorithms here we will require to have the same number of evaluations of the fitness function in each of them.

Theorem 5. *Suppose that the same step mutation operator with a monotone threshold transition matrix is used in the (1+1)-ES and in the (1,λ)-ES. If $Q^{(0)} \geq P^{(0)}$ then $Q^{(t\lambda)} \geq P^{(t)}$ for any t.*

Proof. It is sufficient to give a proof for $t = 1$, since the statement for the general case will follow by induction on t.

a) Consider the moment when k offspring of $b^{(0)}$ have been generated and let $b^{(1,k)}$ denote an offspring with the highest fitness among them. Let us first assume that $b^{(0)}$ and $g^{(0)}$ are fixed so that $b^{(0)} \in H_i \setminus H_{i+1}$ and $g^{(0)} \in H_i \setminus H_{i+1}$ for some i. Then for the (1,λ)-ES for arbitrary $j = 1 \ldots d$ we have:

$$P\{b^{(1,k)} \notin H_j\} = P\{b^{(1,k-1)} \notin H_j\} P\{Mut(b^{(0)}) \notin H_j\}. \tag{5}$$

Now for the (1+1)-ES if $j \leq i$ then $P\{g^{(k)} \notin H_j\} = 0$, and if $j > i$ then

$$P\{g^{(k)} \notin H_j\} = P\{g^{(k-1)} \in H_i \setminus H_j\} \times$$

$$P\{Mut(g^{(k-1)}) \notin H_j | g^{(k-1)} \in H_i \setminus H_j \}. \tag{6}$$

Assume that $P\{b^{(1,k-1)} \notin H_j\} \geq P\{g^{(k-1)} \notin H_j\}$, and thus, $P\{b^{(1,k-1)} \notin H_j\} \geq P\{g^{(k-1)} \in H_i \setminus H_j\}$. From the monotonicity of threshold transition matrix we obtain $P\{Mut(b^{(0)}) \notin H_j\} \geq P\{Mut(g^{(k-1)}) \notin H_j | g^{(k-1)} \in H_i \setminus H_j\}$. Hence, by induction on k, using (5) and (6), we conclude that $P\{b^{(1,k)} \notin H_j\} \geq P\{g^{(k)} \notin H_j\}$ for all $k = 1, ..., \lambda$.

b) To prove that $Q^{(\lambda)} \geq P^{(1)}$ for arbitrary initial distribution with $P^{(0)} = Q^{(0)}$ we use the total probability formula, the step mutation definition and the conclusion of the case a):

$$P\{b^{(1)} \notin H_j\} = \sum_{i=0}^{d} P\{b^{(1)} \notin H_j | b^{(0)} \in H_i \setminus H_{i+1}\}(p_i^{(0)} - p_{i+1}^{(0)}) \geq$$

$$\sum_{i=0}^{d} P\{g^{(\lambda)} \notin H_j | g^{(0)} \in H_i \setminus H_{i+1}\}(q_i^{(0)} - q_{i+1}^{(0)}) = P\{g^{(\lambda)} \notin H_j\}. \tag{7}$$

c) In general, when $Q^{(0)} \geq P^{(0)}$ let us recall that in case of step mutation we can consider $P^{(1)}$ as a function on vector $P^{(0)}$. By (7) we have $Q^{(\lambda)} \geq P^{(1)}(Q^{(0)})$, and due to nonnegativity of the multipliers of probabilities $p_1^{(t-1)}, ..., p_d^{(t-1)}$ in (1), which is an equality in our case, we conclude that $P^{(1)}(Q^{(0)}) \geq P^{(1)}(P^{(0)})$. □

In [3] we showed that the bit-flip mutation operator for the ONEMAX fitness function is an example of a monotone step mutation operator. Analogous result has been obtained there for a family of vertex cover problems of special structure. Of course the situation when we have a monotone step mutation operator is not likely to occur while solving difficult optimization problems, unless we use some "artificial" mutation, especially adjusted for the particular problem instance. For example, if the the set of fitness levels $\Phi_0, ..., \Phi_d$ includes all possible values of the fitness function, then existence of a local optimum (which is not global) in terms of Hamming distance, implies that the bit-flip mutation is not a monotone step mutation operator (with respect to the corresponding subsets $H_1, ..., H_d$).

However the step mutation operators may be considered as convenient theoretical constructions since for any given problem and mutation operator Mut, a monotone step mutation operator Mut' can be constructed to have the threshold transition probabilities equal to the monotone lower bounds of Mut. Then Mut' may be viewed as a realization of the worst-case situation for a given set of lower bounds. In this respect Theorem 5 shows that the best possible lower bound on $P^{(t)}$ for the (1,λ)-ES within our framework is not better than the best lower bound on $Q^{(t)}$ for the (1+1)-ES.

A similar result to Theorem 5 can be also obtained to show that the GA based on mutation and selection does not "outperform" the $(1, \lambda)$-ES in terms of probability vectors if a monotone step mutation operator is used.

3 Some Connections with Complexity Theory

The computational complexity theory has been a source of numerous pessimistic results related to optimization algorithms (see e.g. [6, 7]), in particular, it is shown that there are some classes of combinatorial optimization problems for which even computing good approximate solutions is NP-hard. In this section we are going to consider the a priori lower bounds for transition probabilities with respect to hardness of finding the approximate solutions. While doing this we shall use the notation analogous to the one in [8]. Recall that Σ denotes the alphabet $\{0, 1\}$. By Σ^* we denote the set of all strings with symbols from Σ.

Definition 1. *An NP maximization problem P_{max} is a triple $P_{max} = (I, Sol, f_x)$, where $I \subseteq \Sigma^*$ is the set of instances of P_{max} and:*

1. I is recognizable in polynomial time.

2. Given an instance $x \in I$, $Sol(x) \subseteq \Sigma^*$ denotes the set of feasible solutions of x. Given x and y the decision whether $y \in Sol(x)$ may be done in polynomial time, and there exists a polynomial h such that given any $x \in I$ and $y \in Sol(x)$, $|y| \leq h(|x|)$. (Without loss of generality in what follows we assume that the equality $|y| = h(|x|)$ holds for any $x \in I$ and $y \in Sol(x)$.)

3. Given an instance $x \in I$ and $y \in Sol(x)$, $f_x(y)$ is a positive integer objective function (to be maximized), which is computable in polynomial time.

A similar definition may be given for NP minimization problems, and all the following statements could be properly adjusted for the minimization case as well. Here we will consider the maximization problems for convenience. The optimal objective function value for x is denoted by $f_x^* = \max_{y \in Sol(x)} f_x(y)$.

We have made no specific assumptions concerning the method of encoding the solutions in genotype strings. Given the encoding scheme for the feasible solutions of an NP maximization problem, one can use the string y as a genotype with the fitness function identical to $f_x(y)$ on set $Sol(x)$ and equal to 0 on all other genotypes from $\Sigma^{h(|x|)}$. For simplicity the representation with mapping $y(g) \equiv g$ will be assumed below (the same way we could consider a wider class of representations with the mappings computable in polynomial time).

Definition 2. *Given an instance x of an NP maximization problem and $y \in Sol(x)$, the performance ratio of y with respect to x is $R(x, y) = f_x^*/f_x(y)$.*

Definition 3. *Let P_{max} be an NP maximization problem and let T be an algorithm that for any instance x of P_{max} such that $Sol(x) \neq \emptyset$, returns a feasible solution $T(x)$ in polynomial time. Given a function $r : \mathbf{Z}^+ \to [1, \infty)$, T is called an r-approximation algorithm for P_{max} if the performance ratio of $T(x)$ with respect to x is such that $R(x, T(x)) \leq r(|x|)$. The solution $T(x)$ is called an $r(|x|)$-approximate solution then.*

Producing for any instance $x \in P_{max}$ such solution y that $f_x(y) \geq f_x^*/r(|x|)$ (provided that $Sol(x) \neq \emptyset$) is called *approximation of P_{max} within a factor r*. For example it is proven that the maximum cut problem is NP-hard to approximate within a factor 1.012, approximation of the maximum clique problem within $|V|^{1-\varepsilon}$ is most likely to be impossible (where $|V|$ is the number of vertices in graph and $\varepsilon > 0$ is an arbitrary constant), etc. For more details see e.g. [6].

We will also need the formal definitions of a randomized algorithm and of class BPP of languages recognizable with bounded probability in polynomial time (see e.g. [4]). By a randomized algorithm we mean an algorithm which may be executed by a *probabilistic Turing machine*, i.e. the Turing machine which has a special state for "tossing a coin". When the machine enters this state it receives a bit which is 0 with probability 1/2 and 1 with probability 1/2. A *polynomial-time probabilistic Turing machine* is a probabilistic Turing machine which always halts after a polynomial number of steps.

Definition 4. *BPP is the class of languages $L \subseteq \Sigma^*$ for which there exists a polynomial-time probabilistic Turing machine M, such that:*
1) *For all $x \in L$ holds $P\{M \text{gives an output } 1\} \geq 3/4$.*
2) *For all $x \notin L$ holds $P\{M \text{gives an output } 0\} \geq 3/4$.*

Studying the complexity of mutation operators with certain a priori lower bounds for transition probabilities we will consider the possibility of implementing the mutation by a polynomial-time randomized algorithm. The following supplementary result will be of use.

Lemma 2. *Let P_{max} be an NP maximization problem and the approximation of P_{max} within a factor r is NP-hard. Then unless $NP \subseteq BPP$, no polynomial-time randomized algorithm obtains the r-approximate solutions to all instances x of P_{max} with probability more than $1/\mathrm{poly}(|x|)$, where $\mathrm{poly}(|x|)$ is a polynomial in the length of input x. (The proof is omitted for brevity.)*

The conjecture $NP \not\subseteq BPP$ is equivalent to $NP \neq RP$ and widely believed to be true [9] (here RP denotes the class of languages L recognizable by polynomial-time probabilistic Turing machine with the probability of error equal to zero for inputs $x \notin L$ and not more than $\frac{1}{2}$ if $x \in L$).

Assuming $r = 1$ in Lemma 2 we see that for any NP-hard NP maximization problem most likely there are no randomized polynomial-time algorithms finding the optimum with a probability bounded below by a positive constant.

Using Lemma 2 we can give a pessimistic estimate of the probability of obtaining "good" solutions by a mutation operator computable in polynomial time, when the genotype under mutation represents a relatively "poor" solution.

Theorem 6. *Suppose that P_{max} is an NP maximization problem, the approximation of P_{max} within a factor r is NP-hard, and there is an r'-approximation algorithm for it. Then unless $NP \subseteq BPP$, no polynomial-time mutation operator exists such that for some polynomial $\mathrm{poly}(|x|)$ for all instances x there would be monotone transition bounds with $\Phi_i \leq f_x^*/r'$, $\Phi_j \geq f_x^*/r$ and $\alpha_{ij} \geq 1/\mathrm{poly}(|x|)$.*

Proof. Consider a polynomial-time randomized algorithm which obtains an approximate solution y by the r'-approximation algorithm, and applies the mutation operator to genotype $g = y$. Now since $g \in H_i$ there must exist such k, $i \leq k \leq d$ that $g \in H_k \backslash H_{k+1}$. Due to monotonicity of the transition bounds, $\alpha_{kj} \geq \alpha_{ij} \geq 1/\text{poly}(|x|)$. Thus Lemma 2 yields the required statement. □

For example setting $r = 1$ here we conclude that for any NP-hard NP maximization problem most likely no polynomial-time mutation operator exists to guarantee for all instances x the monotone transition bounds with $\Phi_i \leq f_x^*/r'$, $\Phi_d = f_x^*$ unless α_{id} is a quickly vanishing function of $|x|$.

Theorem 6 suggests that solving the NP-hard problems one should not expect to be able to find polynomial-time mutation operators with monotone lower bounds much better than the guaranteed performance bounds of deterministic polynomial-time algorithms (unless $NP \subseteq BPP$).

4 Conclusions

In this paper we presented the lower and upper bounds for probability of obtaining solutions of sufficient quality for the (1,λ)-ES and the (1+1)-ES evolutionary strategies. In terms of these probabilities the algorithm (1+1)-ES is proved to be always at least as good as the (1,λ)-ES when a step mutation operator with monotone threshold transition matrix is used. Finally, provided that the conjecture $NP \not\subseteq BPP$ holds, the best possible lower bounds on transition probability for polynomial-time mutation operator are shown to approach zero in many cases, when dealing with NP-hard combinatorial optimization problems.

Further research is expected to address the estimation of the average number of iterations until the (1+1)-ES obtains the optimum. Besides that it will be interesting to study some benchmark optimization problems in the framework used in this paper.

References

1. Bäck, T., Schwefel H.-P.: An overview of evolutionary algorithms for parameter optimization, Evolutionary Computation, **1** (1) (1993) 1–23
2. Rechenberg I.: Evolutionsstrategie'94. Formann-Holzboog Verlag, Stuttgart (1994)
3. Eremeev A.V.: Modeling and Analysis of Genetic Algorithm with Tournament Selection. Proc. of Artificial Evolution Conference (AE'99). Lecture Notes in Computer Science, Vol. 1829. Springer Verlag (2000) 84–95
4. Motwani, R. and Raghavan, P.: Randomized Algorithms. Cambridge University Press (1995)
5. Rudolph, G.: Finite Markov Chain Results in Evolutionary Computation: A Tour d'Horizon, Fundamenta Informaticae **35** (1-4) (1998) 67–89
6. Arora, S., Lund, C.: Hardness of approximations. Approximation Algorithms for NP-Hard Problems. Ed. by S.D.Hochbaum. PWS Publishing Company, (1995) 399–446
7. Garey, M.R. and Johnson, D.S.: Computers and intractability. A guide to the theory of NP-completeness. W.H. Freeman and Company, San Francisco (1979)
8. Ausiello, G., Protasi, M.: Local search, reducibility and approximability of NP-optimization problems. Information Processing Letters, **54** (1995) 73–79
9. Ko, K.: Some observations on the probabilistic algorithms and NP-hard problems, Information Processing Letters, **14** (1982) 39–43