Algorithmica manuscript No.
(will be inserted by the editor)

Erratum to: Reoptimization Time Analysis of Evolutionary Algorithms on Linear Functions Under Dynamic Uniform Constraints

Feng Shi · **Martin Schirneck** · **Tobias Friedrich**
Timo Kötzing · **Frank Neumann**

Received: date / Accepted: date

Abstract In the article *Reoptimization Time Analysis of Evolutionary Algorithms on Linear Functions Under Dynamic Uniform Constraints*, we claimed a worst-case runtime of $\mathrm{O}(nD\log D)$ and $\mathrm{O}(nD)$ for the Multi-Objective Evolutionary Algorithm and the Multi-Objective $(\mu+(\lambda,\lambda))$ Genetic Algorithm, respectively, on linear profit functions under dynamic uniform constraint, where $D=|B-B^*|$ denotes the difference between the original constraint bound B and the new one B^*. The technique used to prove these results contained an error. We correct this mistake and show a weaker bound of $\mathrm{O}(nD^2)$ for both algorithms instead.

Acknowledgements We thank Benjamin Doerr and Carola Doerr for spotting the mistake and providing the counter example given below.

Erratum to: DOI 10.1007/s00453-018-0451-4

The Multi-Objective Evolutionary Algorithm

In Theorem 9 of the original article [4], we claimed a bound of $\mathrm{O}(nD\log D)$ for the expected reoptimization time of the Multi-Objective Evolutionary Algorithm (MOEA), where $D=|B-B^*|$ denotes the difference between the original constraint bound B and the new one B^*. Its proof was based on the notion of *candidate* solutions x for which there is an optimum x^* (assuming $B\le B^*$) such that $x_i=1$ implies $x_i^*=1$. In other words, an optimum can be created from a candidate by only flipping 0-bits. We used drift analysis on the potential function $G=B^*-h$, where h is the largest Hamming weight among all candidates in S. The analysis relied on this potential to be non-increasing during the reoptimization. This is not the case as illustrated by the following counterexample.

Suppose $n=5, B=2$, and $B^*=4$; the weights of the profit function P shall observe the inequalities $w_1\ge w_2\ge w_3\ge w_4>w_5$ and $w_2+w_5>w_3+w_4$, the population S may consist of the solutions $y=11000$ and $z=10110$. The unique optimum of P under constraint B^* is $x^*=11110$, making both members of S candidates. Their largest Hamming weight h is 3. A mutation of z flipping 4 bits may result in the string $z'=11001$, which replaces z due to the higher profit. However, z' is not a candidate anymore. The candidate of highest Hamming weight is now y and h decreases to 2. This increases the potential G in turn.

We now prove a weaker runtime bound for the MOEA with an alternative technique.

F. Shi ✉
School of Information Science and Engineering, Central South University, Changsha, Hunan, P.R. China
E-mail: fengshi@mail.csu.edu.cn

M. Schirneck · T. Friedrich · T. Kötzing
Hasso Plattner Institute, Potsdam, Germany

F. Neumann
School of Computer Science, The University of Adelaide, Adelaide, Australia

Theorem 9 *The reoptimization time of the* MOEA *on linear functions under dynamic uniform constraint is*

$$E[T] = O(nD^2).$$

Proof. We first present the analysis for the case of $B \leq B^*$. For any integer u such that $B \leq u < B^*$, let $x^{(u)} = \arg\max_{|x|_1=u} P(x)$ be an arbitrary solution of maximum profit among all solutions with Hamming weight u. Assume, for some $B \leq u < B^*$, $x^{(u)}$ is contained in the population S. Then, choosing $x^{(u)}$ for mutation and flipping exactly one 0-bit of maximum weight and nothing else creates solution $x^{(u+1)}$ within an expected number of at most $en(D+1)$ generations. Observe that $x^{(u)}$ can only be replaced by a solution with the same Hamming weight and at least the same profit, based on the partial order $\succcurlyeq_{\mathrm{MOEA}}$. Thus, the replacement does not influence the expected number of generations that the MOEA needs to get to solution $x^{(u+1)}$. Summing over all $D = |B^* - B|$ waiting times for the $x^{(u+1)}$, starting from the initial solution $x_{\mathrm{orig}} = x^{(B)} \in S$, gives the claimed bound.

The only difference in the case of $B > B^*$ is that the solution $x^{(u)} = \arg\max_{|x|_1=u} P(x)$ is defined for all $B^* < u \leq B$, and we consider the probability choosing $x^{(u)}$ for mutation and flipping exactly one 1-*bit of minimum weight* and nothing else. The rest of the argument is identical. □

Besides, in the original proof of Theorem 12 in [4], regarding the MOEA with single bit flip operator (MOEA-S) on linear functions under dynamic uniform constraints, we used the fact that "[we] pessimistically assume that the optimum $x^* = 1^{B^*}0^{n-B^*}$ is unique, i.e., $w_{B^*} > w_{B^*+1}$." This argument was susceptible of being non-rigorous. We provide a minor revision which shows that the proof remains valid without this assumption.

We now measure the progress of the algorithm in a more general way which includes the special case where the optimum is unique. The weights of the profit function P can be assumed to be ordered non-increasingly. Let B' and B'' be the numbers of bits whose weights are *strictly larger than* w_{B^*} and *at least* w_{B^*}, respectively. This way, we have $B' < B^* \leq B''$, the special case of a unique optimum corresponds to $B^* = B''$. Let x be an arbitrary solution. Define

$$t_{B^*}(x) = |x_{[1,B']}|_1 \quad \text{and} \quad h_{B^*}(x) = |x_{[B'+1,B'']}|_1$$

to be the numbers of 1-bits in x with weights strictly larger than w_{B^*} and exactly w_{B^*}, respectively. Let

$$s_{B^*}(x) = \min\{B^* - B', h_{B^*}(x)\}.$$

We make this distinction since only $B^* - B'$ many of the $B'' - B'$ bits in x with weight w_{B^*} need to be set to 1 to form an optimal solution with Hamming weight B^*. In total, the number of "correctly placed" 1-bits in x compared to an optimal solution with Hamming weight B^* is

$$u_{B^*}(x) = s_{B^*}(x) + t_{B^*}(x).$$

Let y and z denote the two solutions of Hamming weight $B^* - 1$ and B^*, respectively, after the MOEA-S reached the new boundary. Instead of the potential $G = 2B^* - 1 - |y_{[1,B^*]}|_1 - |z_{[1,B^*]}|_1$, we now consider

$$G = 2B^* - 1 - u_{B^*}(y) - u_{B^*}(z),$$

which intuitively measures the number of incorrect 1-bits in z compared to an optimal solution with Hamming weight B^*, but also considers the state of y. As in the original proof, an optimal solution is obtained if $G = 0$.

Consider a 1-bit at position k in z. If either $1 \leq k \leq B'$ holds, or $B' + 1 \leq k \leq B''$ and $h_{B^*}(z) \leq B^* - B'$, then z_k is called *correct*; otherwise, *incorrect*. Let i be the position of the defect between y and z, i.e., $y_i = 0$ and $z_i = 1$. If z_i is correct, then we consider an incorrect 1-bit z_j, there are at least $B^* - u_{B^*}(z)$ many of them. Choosing the solution z for mutation and flipping z_j results in a new solution z' with Hamming weight $B^* - 1$ and profit $P(z') > P(y)$. It thus replaces y and decreases the potential by 1.

Now consider the case that z_i is incorrect. If $h_{B^*}(z) < B^* - B'$ then we consider a 0-bit y_j in y with $1 \leq j \leq B''$, of which there are $B'' - u_{B^*}(y) \geq B^* - u_{B^*}(y)$ many. If $h_{B^*}(z) \geq B^* - B'$, then we let y_j be one of the $B' - s_{B^*}(y) = B^* - u_{B^*}(y)$ 0-bits in y with index $1 \leq j \leq B'$. So, if z_i is incorrect, there are $B^* - u_{B^*}(y)$ many 0-bits in y such that flipping any of them results in a solution y' with $P(y') > P(z)$, replacing z, and also a decrease of the potential.

Putting it all together, we have an expected drift with respect to the potential G of

$$E[G - G'] \geq p\frac{B^* - u_{B^*}(z)}{2n} + (1-p)\frac{B^* - u_{B^*}(y)}{2n} \geq \frac{G}{4n},$$

where p denotes the probability that in the current round the 1-bit z_i is correct. The remaining reasoning for the reoptimization time based on the drift is the same as in the original proof.

Profit Function	(1+1) **EA**	**MOEA**	**MOEA-S**	**MO** $(\mu+(\lambda,\lambda))$ **GA**	
ONEMAX	$O\big(n\log\big(\frac{n-B}{n-B^*}\big)\big)$	$O\big(nD\log\big(\frac{n-B}{n-B^*}\big)\big)$	$O\big(n\log\big(\frac{n-B}{n-B^*}\big)\big)$	$O\Big(\min\{\sqrt{nD^3}, D^2\sqrt{\frac{n}{n-B^*}}\}\Big)$	if $B \le B^*$
	$O\big(n\log\big(\frac{B}{B^*}\big)\big)$	$O\big(nD\log\big(\frac{B}{B^*}\big)\big)$	$O\big(n\log\big(\frac{B}{B^*}\big)\big)$	$O\Big(\min\{\sqrt{nD^3}, D^2\sqrt{\frac{n}{B^*}}\}\Big)$	if $B > B^*$
linear function	$O\big(n^2\log(B^* w_{\max})\big)$	$O\big(nD^2\big)$	$O(n\log D)$	$O\big(nD^2\big)$	

Table 1: Overview of Results. Upper bounds on the expected reoptimization times of the (1+1) EA, the Multi-Objective Evolutionary Algorithm (MOEA), its variant with single bit flip (MOEA-S) and the Multi-Objective $(\mu+(\lambda,\lambda))$ Genetic Algorithm (MO $(\mu+(\lambda,\lambda))$ GA) on linear functions of length-n bit strings under dynamic uniform constraint. B denotes the old and B^* the new cardinality bound, $D = |B^* - B|$ their difference. Runtimes of the form $O(n\log(B/B^*))$ are to be read as $O(n\log B)$ if $B^* = 0$. For comparison, the (1+1) EA needs $\Omega(n)$ iterations to optimize ONEMAX under uniform constraint from scratch in the static setting (if B is not too close to 0, n or $n/2$) and $\Omega(n^2)$ for general linear profit functions [3].

The Multi-Objective $(\mu+(\lambda,\lambda))$ Genetic Algorithm

The analysis of the Multi-Objective $(\mu+(\lambda,\lambda))$ Genetic Algorithm (MO $(\mu+(\lambda,\lambda))$ GA) on general linear profit functions in the original article [4] had the same error as Theorem 9. In the following, we restate the corrected Subsection 5.2 as a whole for completeness. Additionally, the summary of the results in Table 1 were also updated, both for the MOEA and the MO $(\mu+(\lambda,\lambda))$ GA on linear functions the bound is now $O(nD^2)$.

5.2 Linear Function with Dynamic Uniform Constraint

We turn to linear functions under dynamic uniform constraints and start by lower bounding the probability of an improvement if the parameters are set in a right way. Similar to the analysis for ONEMAX, we study the probability of an iteration of the while-loop in the MO $(\mu+(\lambda,\lambda))$ GA to find an optimal solution with Hamming weight $A+1$ (if $B \le B^*$; $A-1$ if $B > B^*$) starting with an optimal solution x of Hamming weight A, where optimality is defined with respect to a linear profit function. As the bits may have different weights, we need to consider a 0-bit in x yielding the maximum profit increase if $B \le B^*$ (resp., a 1-bit yielding the minimum profit decrease if $B > B^*$) to get an optimal solution with Hamming weight $A+1$ (resp., $A-1$). Thus, the variable λ is fixed at value $\lceil\sqrt{n}\,\rceil$ throughout this subsection, and not chosen depending on the fitness of the current solution.

The following two lemmata are adaptations of the related results given in [2] to the MO $(\mu+(\lambda,\lambda))$ GA working on linear profit functions.

Lemma 16 *Consider the* MO $(\mu+(\lambda,\lambda))$ GA *working on a linear profit function under dynamic uniform constraint $B \le B^*$. Assume its parameters are set as $\lambda = \lceil\sqrt{n}\,\rceil$, $p = \lambda/n$, and $c = 1/\lambda$. When choosing an optimal solution x with $B \le |x|_1 < B^*$ for reproduction, the probability of an iteration of the while-loop to produce an optimal solution y^* with $|y^*|_1 = |x|_1 + 1$ is at least C/λ, with $C > 0$ a constant.*

Proof. The reasoning is similar to that of the proof of [4, Lemma 13]. To get an optimal solution y^* with Hamming weight $|x|_1 + 1$, it is necessary that the solution x' obtained in the mutation phase has $x'_j = 1$, where j is the index of a 0-bit with maximum weight in x. Recall that ℓ is the number of bits to flip, which is drawn according to a binomial distribution with parameters n and p at the beginning of the mutation phase. For λ mutants, the probability that at least one of them has its j-th bit set to 1 is at least $1 - \left(1 - \frac{\ell}{n}\right)^{\lambda}$. Note that the mutant whose j-th bit is flipped to 1 may not be the unique valid offspring of x among the λ mutants, thus the solution is chosen as x' with probability $\Omega(1/\lambda)$, and the event $x'_j = 1$ happens at the end of the mutation phase with probability $\frac{1}{\lambda}\left(1 - \left(1 - \frac{\ell}{n}\right)^{\lambda}\right)$. Combining the above probability and the analysis given for the crossover phase in Lemma 13, we get that an iteration of the while-loop gets an optimal solution y^* having Hamming weight $|x|_1 + 1$ with probability at least

$$\frac{1}{\lambda}\left(1-\left(1-\frac{\ell}{n}\right)^{\lambda}\right)\left(1-\left(1-c\,(1-c)^{\ell-1}\right)^{\lambda}\right).$$

Now we give a lower bound for this probability. Recall that $\lambda = \lceil \sqrt{n} \rceil$, so for sufficiently large n, we have $n \geq 7\lambda/4$. Let $L \sim \mathrm{Bin}(n,p)$ be a binomially distributed random variable, and K be the indicator variable of sampling an optimal solution y^* having Hamming weight $|x|_1 + 1$ within one iteration of the while-loop. From the law of total probability, we get

$$\Pr[K] \geq \sum_{\ell=\lceil \lambda/4 \rceil}^{7\lambda/4} \Pr[K|L=\ell] \cdot \Pr[L=\ell],$$

where $\Pr[K|L=\ell] \geq \frac{1}{\lambda}(1-\left(1-\frac{\ell}{n}\right)^{\lambda})(1-(1-c(1-c)^{\ell-1})^{\lambda})$ as described above.

We bound the term $1-\left(1-\frac{\ell}{n}\right)^{\lambda}$ from below using that $\ell \geq \lceil \lambda/4 \rceil$ and $\lambda = \lceil \sqrt{n} \rceil$,

$$1-\left(1-\frac{\ell}{n}\right)^{\lambda} \geq 1-\left(1-\frac{\lambda/4}{n}\right)^{\lambda} = 1-\left(1-\frac{1}{4\sqrt{n}}\right)^{\sqrt{n}} \geq 1-e^{-\frac{1}{4}}.$$

We have already shown in Lemma 13 (in [4]) that the term $1-(1-c(1-c)^{\ell-1})^{\lambda}$ is at least $1-e^{-\frac{1}{8\sqrt{2}}}$. Thus, $\Pr[K|L=\ell] \geq \alpha/\lambda$, where $\alpha = (1-e^{-\frac{1}{4}})(1-e^{-\frac{1}{8\sqrt{2}}})$ is a positive constant, holds for the desired range of ℓ.

Inserting this back into above inequality gives

$$\Pr[K] \geq \sum_{\ell=\lceil \lambda/4 \rceil}^{7\lambda/4} \Pr[K|L=\ell] \cdot \Pr[L=\ell] \geq \frac{\alpha}{\lambda} \cdot \sum_{\ell=\lceil \lambda/4 \rceil}^{7\lambda/4} \Pr[L=\ell] = \frac{\alpha}{\lambda} \cdot \Pr[\lambda/4 \leq L \leq 7\lambda/4].$$

Note that the mutation probability p is set to λ/n, hence $E[L] = np = \lambda$. Then, $\lambda/4 \leq L \leq 7\lambda/4$ describes the event that L has at most a $1 \pm 3/4$ relative deviation from its mean. By Chernoff bounds [1], this has probability exponentially close to 1, in particular it is bounded below by a constant for all n large enough. Combining the arguments from above discussion now shows that there exists a constant $C > 0$ such that $\Pr[K] \geq C/\lambda$. □

The proof needs only minor changes to accommodate for $B > B^*$. The index j now marks a 1-bit of minimum weight in x and the binary variable K indicates the event of sampling an optimal solution y^* with $|y^*|_1 = |x|_1 - 1$ within one iteration of the while-loop. The reasoning over these variables stays exactly the same. From this, we get the following lemma.

Lemma 17 *Consider the* MO $(\mu+(\lambda,\lambda))$ GA *working on a linear profit function under dynamic uniform constraint $B > B^*$. Assume its parameters are set as $\lambda = \lceil \sqrt{n}\, \rceil$, $p = \lambda/n$, and $c = 1/\lambda$. When choosing an optimal solution x with $B \geq |x|_1 > B^*$ for reproduction, the probability of an iteration of the while-loop to produce an optimal solution y^* with $|y^*|_1 = |x|_1 - 1$ is at least C/λ, with $C > 0$ a constant.*

Finally, we show the upper bound of the expected reoptimization time for the MO $(\mu+(\lambda,\lambda))$ GA on linear functions with dynamic uniform constraints.

Theorem 18 *Consider the* MO $(\mu+(\lambda,\lambda))$ GA *with a static parameter setting of $\lambda = \lceil \sqrt{n}\, \rceil$, mutation probability $p = \lambda/n$, and crossover probability $c = 1/\lambda$. Its expected reoptimization time on linear profit functions under dynamic uniform constraint is* $\mathrm{O}(nD^2)$.

Proof. We show the expected reoptimization time in a similar way to the proof of [4, Theorem 15], first for $B \leq B^*$. For any $B \leq A < B^*$, let $x^{(A)}$ denote a solution of maximum profit among all solutions in the search space with Hamming weight A. Assume there is an A such that $x^{(A)}$ is in the population S. Note that A may not be the maximum Hamming weight of solutions in S and the size of S cannot be bounded by $A - B + 1$, as was the case in Theorem 15. We only have a weaker bound of $|S| \leq B^* - B + 1 = D + 1$.

Now, combining Lemma 16 and the fact that the probability of choosing $x^{(A)}$ for reproduction is in $\Omega(1/D)$ implies that the MO $(\mu+(\lambda,\lambda))$ GA takes an expected number of $\mathrm{O}(D\lambda^2) = O(nD)$ fitness evaluations to find an optimal solution with Hamming weight $A+1$. By summing over the waiting times for all Hamming weights between B and B^* starting with the initial solution $x_{\mathrm{orig}} = x^{(B)}$, we get an expected reoptimization time in $O(nD^2)$. Via Lemma 17, the adaption to $B > B^*$ is immediate. □

References

1. Doerr, B.: Analyzing randomized search heuristics: Tools from probability theory. In: Theory of Randomized Search Heuristics: Foundations and Recent Developments, pp. 1–20. World Scientific (2011)
2. Doerr, B., Doerr, C., Ebel, F.: From black-box complexity to designing new genetic algorithms. Theoretical Computer Science **567**, 87–104 (2015)
3. Friedrich, T., Kötzing, T., Lagodzinski, J.A.G., Neumann, F., Schirneck, M.: Analysis of the (1+1) EA on subclasses of linear functions under uniform and linear constraints. In: Proceedings of the Workshop on the Foundations of Genetic Algorithms (FOGA), pp. 45–54. ACM (2017)
4. Shi, F., Schirneck, M., Friedrich, T., Kötzing, T., Neumann, F.: Reoptimization time analysis of evolutionary algorithms on linear functions under dynamic uniform constraints. Algorithmica **81**(2), 828–857 (2019)