Truncation Selection and Gaussian EDA: Bounds for Sustainable Progress in High-Dimensional Spaces

Petr Pošík

Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Technická 2, 166 27 Prague 6, Czech Republic
posik@labe.felk.cvut.cz

Abstract. In real-valued estimation-of-distribution algorithms, the Gaussian distribution is often used along with maximum likelihood (ML) estimation of its parameters. Such a process is highly prone to premature convergence. The simplest method for preventing premature convergence of Gaussian distribution is enlarging the maximum likelihood estimate of σ by a constant factor k each generation. Such a factor should be large enough to prevent convergence on slopes of the fitness function, but should not be too large to allow the algorithm converge in the neighborhood of the optimum. Previous work showed that for truncation selection such admissible k exists in 1D case. In this article it is shown experimentaly, that for the Gaussian EDA with truncation selection in high-dimensional spaces no admissible k exists!

1 Introduction

Estimation of Distribution Algorithms (EDAs) [1] are a class of Evolutionary Algorithms (EAs) that do not use the crossover and mutation operators to create the offspring population. Instead, they build a probabilistic model describing the distribution of selected individuals and create offspring by sampling from the model.

In EDAs working in real domain (real-valued EDAs), the Gaussian distribution is often employed as the model of promissing individuals ([2], [3], [4], [5]). The distribution is often learned by maximum likelihood (ML) estimation. It was recognized by many authors (see e.g. [3], [6], [7]) that such a learning scheme makes the algorithm very prone to premature convergence: in [8], it was shown also theoretically that the distance that can be traversed by a simple Gaussian EDA with truncation selection is bounded, and [9] showed similar results for tournament selection.

Many techniques that fight the premature convergence were developed, usually by means of artificially enlarging the ML estimate of variance of the learned distribution. In [6], the variance is kept on values greater than 1, while [7] used self-adaptation of the variance. Adaptive variance scaling (AVS), i.e. enlarging

M. Giacobini et al. (Eds.): EvoWorkshops 2008, LNCS 4974, pp. 525–534, 2008.

the variance when better solutions were found and shrinking the variance in case of no improvement, was used along with various techniques to trigger the AVS only on the slope of the fitness function in [10] and [11].

However, in the context of the simple Gaussian EDAs there exists a much simpler method for preventing premature convergence (compared to the above mentioned techniques): enlarging the ML estimate of standard deviation each generation by a factor k that is held constant during the whole evolution. The resulting EDA is depicted in Fig. 1. Although the parameter learning schemes suggested in the works mentioned above are more sophisticated, the approach studied in this article is appealing mainly from the *simplicity* point of view. To the best of the author's knowledge, nobody has shown yet that this approch does not work.

```
1. Set the initial values of parameters $\mu^0 = (\mu_1^0, \dots, \mu_D^0)$ and $\sigma^0 = (\sigma_1^0, \dots, \sigma_D^0)$, $D$ is the dimensionality of the search space. Set the generation counter $t = 0$.
2. Sample $N$ new individuals from the distribution $\mathcal{N}(\mu^t, \sigma^t I_D)$, $I_D$ is the identity matrix.
3. Evaluate the individuals.
4. Select the $\tau N$ best solutions.
5. Estimate new values of parameters $\mu^{t+1}$ and $\sigma^{t+1}$ based on the selected individuals independently for each dimension.
6. Enlarge the $\sigma^{t+1}$ by a constant factor $k$.
7. Advance generation counter: $t = t + 1$.
8. If termination condition is not met, go to step 2.
```

Fig. 1. Simple Gaussian EDA analysed in this article

The algorithm in Fig. 1 was studied in [12] where the minimal values of the 'amplification coefficient' were determined by search in 1D case. In [13], the theoretical model of the algorithm behavior in 1D was used to derive the minimal and maximal admissible values for k. In this paper, the behaviour of the algorithm in multidimensional case is studied with the aim of developing the bounds for values of k if they exist.

The rest of the paper is organized as follows: in Sec. 2, the relevant results from [13] are surveyed. Section 3 presents the experimental methodology used to determine the bounds for k in more-dimensional spaces, the results of the experiments and their discussion. Finally, Sec. 4 summarizes and concludes the paper, and presents some pointers to future work.

2 Fundamental Requirements on EDA

When analysing the behaviour of evolutionary algortithms, the fitness landscape is often modelled as consisting of slopes and valleys (see e.g. [10], [14], [12]).

There are two fundamental requirements on the development of the population variance that ensure reasonable behavior of the algorithm as a whole:

1. The variance *must not shrink on the slope.* This ensures that the population position is not bounded and that it eventually finds at least a local optimum.
2. The variance *must shrink in the valley.* In the neighborhood of the optimum, the algorithm must be allowed to converge to find the optimum precisely.

These two conditions constitute the bounds for the variance enlarging constant k which must be large enough to traverse the slopes, but must not be too large to focus to the optimum. In this article, the slopes are modelled with a linear function (Eq. 1) that takes into account only the first coordinate of the individual, and the valleys are modelled as a quadratic bowl (Eq. 2):

$$f_{\text{lin}}(\mathbf{x}) = x_1 \tag{1}$$

$$f_{\text{quad}}(\mathbf{x}) = \sum_{d=1}^{D} x_d^2 \tag{2}$$

Since the fitness function influences the algorithm only by means of the truncation selection, the actual values of the fitness function do not matter, only their order. In 1D space, the results derived for the linear function thus hold for all monotonous functions, in more-dimensional spaces for all functions with parallel contour lines and monotonous function values along the gradient. Similarly, the results derived for the quadratic function also hold for any unimodal function isotropic around its optimum with function values monotonously increasing with the distance from the optimum.

2.1 Bounds for k

In this paper, it is assumed that the development of variance can be modelled with the following recurrent equation:

$$(\sigma^{t+1})^2 = (\sigma^t)^2 \cdot c, \tag{3}$$

where c is the ratio of the population variances in two consecutive generations, t and $t+1$.

As already said in the introduction, the simplest method of preventing premature convergence is to enlarge the estimated standard deviation σ by a constant factor k. Thus

$$\sigma^{t+1} = k \cdot \sigma^t \cdot \sqrt{c} \tag{4}$$

In order to prevent the premature convergence on the slope, the ratio of the consecutive standard deviations should be at least 1, i.e.

$$\frac{\sigma^{t+1}}{\sigma^t} = k \cdot \sqrt{c_{\text{slope}}} \geq 1 \tag{5}$$

$$k \geq \frac{1}{\sqrt{c_{\text{slope}}}} \overset{\text{def}}{=} k_{\min} \tag{6}$$

On the other hand, to be able to focus to the optimum, the model must be allowed to converge in the valley. The ratio of the two consecutive standard deviations should be lower than 1, i.e.

$$\frac{\sigma^{t+1}}{\sigma^t} = k \cdot \sqrt{c_{\text{valley}}} < 1 \tag{7}$$

$$k < \frac{1}{\sqrt{c_{\text{valley}}}} \overset{\text{def}}{=} k_{\max} \tag{8}$$

Joining these two conditions together gives us the bounds for the constant k:

$$k_{\min} = \frac{1}{\sqrt{c_{\text{slope}}}} \leq k < \frac{1}{\sqrt{c_{\text{valley}}}} = k_{\max} \tag{9}$$

In this paper, the value of k is called *admissible* if it satisfies condition 9.

2.2 Model of EDA Behaviour in 1D

For 1D case, the bounds of Eq. 9 are known and can be computed theoretically (see [13]). The behavior of the simple Gaussian EDA with truncation selection in 1D space can be modelled by using statistics for the truncated normal distribution ([13], [10]). In one iteration of the EDA, the variance of the population changes in the following way:

$$(\sigma^{t+1})^2 = (\sigma^t)^2 \cdot c(z_1, z_2) \tag{10}$$

where

$$c(z_1, z_2) = 1 - \frac{z_2 \cdot \phi(z_2) - z_1 \cdot \phi(z_1)}{\Phi(z_2) - \Phi(z_1)} - \left(\frac{\phi(z_2) - \phi(z_1)}{\Phi(z_2) - \Phi(z_1)}\right)^2, \tag{11}$$

$\phi(z)$ and $\Phi(z)$ are the probability density function (PDF) and cumulative distribution function (CDF) of the standard normal distribution $\mathcal{N}(0, 1)$, respectively, and z_i are the truncation points (measured in the standard deviations) which are constant during the EDA run for both f_{lin} and f_{quad}.

The effect of the truncation selection on the population distribution is different depending on the following situations:

- The population is on the slope of the fitness function.
- The population is in the valley of the fitness function.

First, suppose *the population is on the slope* of the fitness function (which is modelled by linear function). The best $\tau \cdot N$ individuals are at the left-hand side of the distribution, i.e.

$$z_1 \to -\infty, \text{ and } z_2 = \Phi^{-1}(\tau), \tag{12}$$

where $\Phi^{-1}(\tau)$ is the inverse cumulative distribution function of the standard normal distribution. We can thus define

$$c_{\text{slope}}(\tau) = c\left(-\infty, \Phi^{-1}(\tau)\right). \tag{13}$$

In the second case, when *the population is centered around the optimum*, the selected $\tau \cdot N$ individuals are centered around the mean of the distribution, thus

$$z_1 = \Phi^{-1}\left(\frac{1-\tau}{2}\right), \text{ and } z_2 = \Phi^{-1}\left(\frac{1+\tau}{2}\right). \tag{14}$$

and we can again define

$$c_{\text{valley}}(\tau) = c\left(\Phi^{-1}\left(\frac{1-\tau}{2}\right), \Phi^{-1}\left(\frac{1+\tau}{2}\right)\right). \tag{15}$$

The above equations are taken from [13], but were already presented (in a bit different form) in previous works (Eq. 13 e.g. in [8] and Eq. 15 in [14]).

The bounds for k in 1D case computed using equations 9, 13, and 15 are depicted on Fig. 2, the same values are shown in tabular form in Table 1.

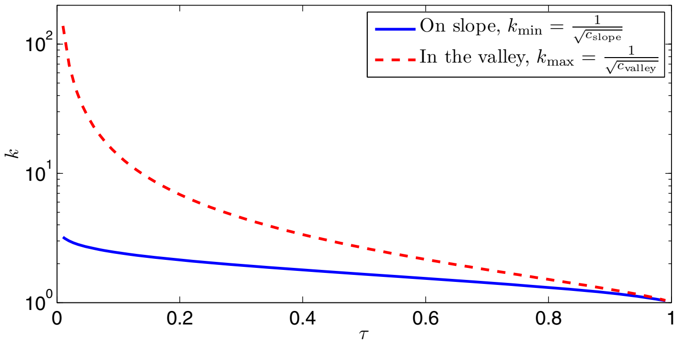

Fig. 2. Minimum and maximum values for setting the enlarging parameter k for various values of τ

Table 1. The bounds for the standard deviation enlargment constant k for various values of τ in 1D case

τ	0.01	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	0.99
k_{min}	3.213	2.432	2.139	1.944	1.791	1.659	1.539	1.424	1.310	1.185	1.033
k_{max}	138.195	13.798	6.866	4.540	3.364	2.648	2.159	1.797	1.511	1.267	1.040

These results show that in 1D case for each value of the selection proportion τ, there is an interval of admissible values of k ensuring that the algorithm will not converge prematurely on the slope of the fitness function, and will be able to focus (i.e. decrease the variance) in the neighborhood of the optimum. It is thus highly appealing to study these features in more-dimensional case.

3 Bounds for k in Multidimensional Space

To derive bounds for k in multidimensional case theoretically (as in case of 1D space) is much more complicated. In this article, an experimental approach is used. The lower bound is found by using the linear fitness function. During the experiments, the value of standard deviation of coordinate x_1 is tracked and it is checked if it increases or decreases (on average). The bisection method is used to determine the value of k for which the variance stays the same (with certain tolerance). Similarly, the upper bound is found by experiments with quadratic function.

3.1 Experimental Methodology

The population size 1,000 was used in all experiments. To determine each particular $k_{\min}$ (and $k_{\max}$), 10 independent runs of 100 generations were carried out. During each run, the standard deviation of x_1 was tracked; this gives 10 values of st.d. for each of 100 generations. Examples of the development of σ when searching for $k_{\max}$ in 5-dimensional space with $\tau = 0.5$ (i.e. using 5D quadratic fitness) can be seen in Fig. 3.

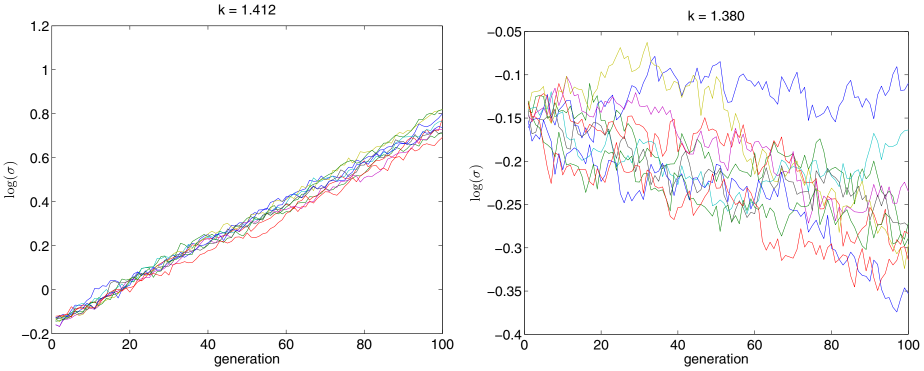

Fig. 3. Examples of the development of standard deviation during the search for $k_{\max}$ in 5-dimensional space with $\tau = 0.5$. On the left: k is by far higher than $k_{\max}$. On the right: k is slightly lower than $k_{\max}$.

To this data, a linear function of the form $E(\log(\text{st.d.})) = a \cdot \text{gen} + b$ was fitted ('gen' is the generation counter) using simple linear regression which should be adequate type of model. The sign of the learned parameter a was used to decide, if the variances increase or decrease during the run. To find the critical value of k (for which the parameter a changes its sign), the bisection method was used with the precision given by 24 iterations.

3.2 Results and Discussion

The minimal and maximal admissible values of the variance enlarging factor k are displayed in Tab. 2 in tabular form, and in Fig. 4 in graphical form.

Table 2. Minimum and maximum admissible values for the enlarging factor k for various values of τ and dimensions 1, …, 6

		τ										
Dim		0.01	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99
1	$k_{\min}$	3.477	2.443	2.140	1.947	1.794	1.660	1.539	1.423	1.309	1.184	1.033
	$k_{\max}$	130.911	13.755	6.862	4.540	3.363	2.647	2.157	1.796	1.510	1.266	1.040
2	$k_{\min}$	3.501	2.443	2.139	1.947	1.791	1.660	1.540	1.424	1.310	1.186	1.033
	$k_{\max}$	14.239	4.396	3.049	2.441	2.068	1.804	1.603	1.438	1.293	1.159	1.024
3	$k_{\min}$	3.520	2.439	2.141	1.946	1.791	1.659	1.540	1.424	1.309	1.185	1.033
	$k_{\max}$	6.667	2.976	2.300	1.958	1.734	1.568	1.434	1.320	1.218	1.120	1.018
4	$k_{\min}$	3.411	2.438	2.142	1.945	1.790	1.660	1.539	1.425	1.310	1.185	1.034
	$k_{\max}$	4.607	2.432	1.981	1.740	1.576	1.452	1.349	1.260	1.179	1.099	1.015
5	$k_{\min}$	3.509	2.442	2.147	1.946	1.794	1.660	1.540	1.424	1.310	1.185	1.033
	$k_{\max}$	3.672	2.145	1.803	1.614	1.485	1.382	1.297	1.223	1.154	1.085	1.013
6	$k_{\min}$	3.481	2.449	2.141	1.948	1.788	1.659	1.539	1.424	1.310	1.184	1.033
	$k_{\max}$	3.140	1.966	1.690	1.533	1.422	1.336	1.262	1.198	1.137	1.076	1.011

Comparing the theoretical values of $k_{\min}$ and $k_{\max}$ from Tab. 1 and experimental values from Tab. 2 for 1D case, we can see that they do not differ substantially which constitutes at least partial confirmation that the model is in accordance with experiments.

Another (expectable) observation is the fact that the minimal as well as maximal bound for k decreases with decreasing selection pressure (with increasing selection proportion τ). It is worth to note, that the lower bound of k does not change with dimensionality (the solid lines in Fig. 4 are almost the same) since only the first coordinate is taken into account in the selection process.

The upper bound of k, however, changes substantially with the dimensionality of the search space. In 1D case, the upper bounds of k for all selection proportions τ lie above the lower bounds, and it is thus possible to find admissible k for every τ. With increasing dimensionality, the upper bound falls below the lower bound for increasingly larger interval of τ (which means that not for every τ we are able to find admissible k). For dimensionality 5 and above, admissible k actually does not exist regardless of τ!

The fact that $k_{\max}$ drops with increasing dimensionality can be explained as follows. Consider the quadratic bowl f_{quad} as the fitness function and the situation when the population is centered around origin. In that case, the individuals selected by truncation selection are bounded by a D-dimensional hypersphere. Its radius (the maximum distance from origin) can be computed as $r = \sqrt{CDF^{-1}_{\chi^2}(\tau, D)}$, where $CDF^{-1}_{\chi^2}(\tau, D)$ is the inverse cumulative distribution function of χ^2 distribution with D degrees of freedom and τ is the selection proportion. As can be seen in Fig. 5 for $\tau = 0.5$, the hypersphere radius grows with dimensionality. It means that the data points after selection are more spread

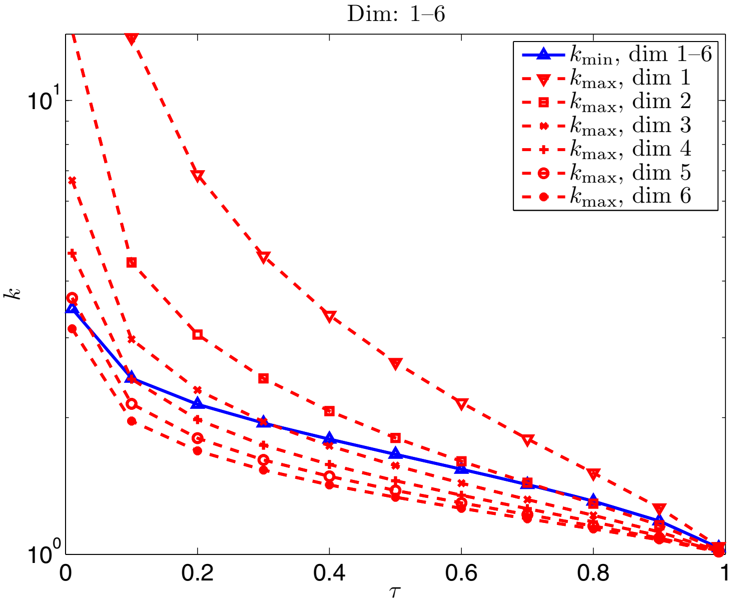

Fig. 4. Minimum and maximum admissible values for the enlarging factor k for various values of τ and dimensions 1, ..., 6, graphically

along the axes and the ML estimate of the standard deviation σ is higher. It is thus necessary to multiply the ML estimate of σ with a lower value of k only, not to overshoot the value 1.

4 Summary and Conclusions

This paper analysed some of the convergence properties of a simple EDA based on Gaussian distribution and truncation selection. Specifically, the simplest way of preventing premature convergence was studied: each generation, the ML estimate of the population standard deviation σ was enlarged by a factor k held constant during the whole evolution. Previous theoretical results for 1D case suggested there should exist an interval of admissible values of k that are

- sufficiently large to allow the algorithm to traverse the slope of the fitness function with at least nondecreasing velocity, and
- sufficiently low to allow the algorithm to exploit the neighborhood of the optimum once it is found.

However, as the experimental results in more dimensional spaces suggest, *the interval of admissible values of k gets smaller with increasing dimensionality and eventually vanishes*! In other words, in more-dimensional spaces there is no single value of the variance enlarging factor k that would allow the simple Gaussian

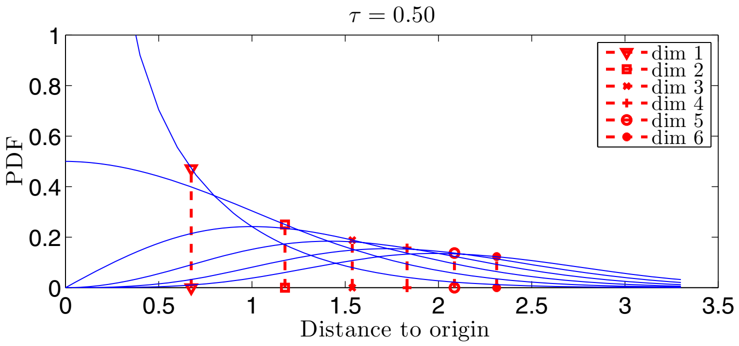

Fig. 5. The radius of hypersphere containing τN best individuals increases with dimensionality. Blue lines: probability density functions (PDF) of the distribution of distances of points sampled from D-dimensional standard normal distribution. Red dashed lines: the hypersphere radiuses for dimensions $1, \ldots, 6$.

EDA with truncation selection to behave reasonably on slopes and in the valleys of the fitness function in the same time!

This result is important on its own, but also serves as another supporting evidence to favor dynamic, adaptive, or self-adaptive control strategies for the standard deviation of the sampling distribution (e.g. those mentioned in Sec. 1), when truncation selection is used. Nevertheless, if the constant enlarging factor should be used, then I suggest at least 2-stage search procedure: in the first stage, set the factor $k > k_{\min}$, so that the optimum is localized; after a few iterations in which the progress is slow, switch to the second phase and set the factor $k < k_{\max}$ (or use directly the ML estimate of σ) to fine tune the solution.

Similar study of other selection mechanisms remains as the future work. If the interval of admissible values exists for them, having the bounds for the factor k is very profitable from the experimenter's and EA designer's point of view. These bounds could be also used in various adaptive variance scaling schemes as safeguards.

Acknowledgments. The author is supported by the Grant Agency of the Czech Republic with the grant no. 102/08/P094 entitled "Machine learning methods for solution construction in evolutionary algorithms".

References

1. Larrañaga, P., Lozano, J.A. (eds.): Estimation of Distribution Algorithms. GENA. Kluwer Academic Publishers, Dordrecht (2002)
2. Larrañaga, P., Lozano, J.A., Bengoetxea, E.: Estimation of distribution algorithms based on multivariate normal distributions and gaussian networks. Technical Report KZZA-IK-1-01, Dept. of Computer Science and Artificial Intelligence, University of Basque Country (2001)

3. Bosman, P.A.N., Thierens, D.: Expanding from discrete to continuous estimation of distribution algorithms: The IDEA. In: Deb, K., Rudolph, G., Lutton, E., Merelo, J.J., Schoenauer, M., Schwefel, H.-P., Yao, X. (eds.) PPSN 2000. LNCS, vol. 1917, pp. 767–776. Springer, Heidelberg (2000)
4. Rudlof, S., Köppen, M.: Stochastic hill climbing by vectors of normal distributions. In: First Online Workshop on Soft Computing, Nagoya, Japan (1996)
5. Ahn, C.W., Ramakrishna, R.S., Goldberg, D.E.: Real-coded bayesian optimization algorithm: Bringing the strength of BOA into the continuous world. In: Deb, K., et al. (eds.) GECCO 2004. LNCS, vol. 3103, pp. 840–851. Springer, Heidelberg (2004)
6. Yuan, B., Gallagher, M.: On the importance of diversity maintenance in estimation of distribution algorithms. In: Beyer, H.G., O'Reilly, U.M. (eds.) Proceedings of the Genetic and Evolutionary Computation Conference GECCO-2005, vol. 1, pp. 719–726. ACM Press, New York (2005)
7. Koumoutsakos, P., Očenášek, J., Hansen, N., Kern, S.: A mixed bayesian optimization algorithm with variance adaptation. In: Yao, X., Burke, E.K., Lozano, J.A., Smith, J., Merelo-Guervós, J.J., Bullinaria, J.A., Rowe, J.E., Tiňo, P., Kabán, A., Schwefel, H.-P. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 352–361. Springer, Heidelberg (2004)
8. Grahl, J., Minner, S., Rothlauf, F.: Behaviour of UMDAc with truncation selection on monotonous functions. In: IEEE Congress on Evolutionary Computation, CEC 2005, vol. 3, pp. 2553–2559 (2005)
9. Gonzales, C., Lozano, J., Larranaga, P.: Mathematical modelling of UMDAc algorithm with tournament selection. International Journal of Approximate Reasoning 31(3), 313–340 (2002)
10. Grahl, J., Bosman, P.A.N., Rothlauf, F.: The correlation-triggered adaptive variance scaling IDEA. In: Proceedings of the 8th annual conference on Genetic and Evolutionary Computation Conference - GECCO 2006, pp. 397–404. ACM Press, New York (2006)
11. Bosman, P.A.N., Grahl, J., Rothlauf, F.: SDR: A better trigger for adaptive variance scaling in normal EDAs. In: GECCO 2007: Proceedings of the 9th annual conference on Genetic and Evolutionary Computation, pp. 492–499. ACM Press, New York (2007)
12. Yuan, B., Gallagher, M.: A mathematical modelling technique for the analysis of the dynamics of a simple continuous EDA. In: IEEE Congress on Evolutionary Computation, CEC 2006, Vancouver, Canada, pp. 1585–1591. IEEE Press, Los Alamitos (2006)
13. Pošík, P.: Gaussian EDA and truncation selection: Setting limits for sustainable progress. In: IEEE SMC International Conference on Distributed Human-Machine Systems, DHMS 2008, Athens, Greece, IEEE, Los Alamitos (2008)
14. Grahl, J., Bosman, P.A.N., Minner, S.: Convergence phases, variance trajectories, and runtime analysis of continuous EDAs. In: GECCO 2007: Proceedings of the 9th annual conference on Genetic and evolutionary computation, pp. 516–522. ACM Press, New York (2007)