J. Hlaváčová (Ed.): ITAT 2017 Proceedings, pp. 120–128
CEUR Workshop Proceedings Vol. 1885, ISSN 1613-0073, © 2017 Z. Pitra, L. Bajer, J. Repický, M. Holeňa

Adaptive Doubly Trained Evolution Control for the Covariance Matrix Adaptation Evolution Strategy

Zbyněk Pitra[1,2,3], Lukáš Bajer[3,4], Jakub Repický[3,4], and Martin Holeňa[3]

[1] National Institute of Mental Health
Topolová 748, 250 67 Klecany, Czech Republic
z.pitra@gmail.com
[2] Faculty of Nuclear Sciences and Physical Engineering, Czech Technical University in Prague
Břehová 7, 115 19 Prague 1, Czech Republic
[3] Institute of Computer Science, Academy of Sciences of the Czech Republic
Pod Vodárenskou věží 2, 182 07 Prague 8, Czech Republic
bajeluk@gmail.com, j.repicky@gmail.com, holena@cs.cas.cz
[4] Faculty of Mathematics and Physics, Charles University in Prague
Malostranské nám. 25, 118 00 Prague 1, Czech Republic

Abstract: An area of increasingly frequent applications of evolutionary optimization to real-world problems is continuous black-box optimization. However, evaluating real-world black-box fitness functions is sometimes very time-consuming or expensive, which interferes with the need of evolutionary algorithms for many fitness evaluations. Therefore, surrogate regression models replacing the original expensive fitness in some of the evaluated points have been in use since the early 2000s. The Doubly Trained Surrogate Covariance Matrix Adaptation Evolution Strategy (DTS-CMA-ES) represents a surrogate-assisted version of the state-of-the-art algorithm for continuous black-box optimization CMA-ES. The DTS-CMA-ES saves expensive function evaluations through using a surrogate model. However, the model inaccuracy on some functions can slow-down the algorithm convergence. This paper investigates an extension of DTS-CMA-ES which controls the usage of the model according to the model's error. Results of testing an adaptive and the original version of DTS-CMA-ES on the set of noiseless benchmarks are reported.

1 Introduction

Evolutionary algorithms have become very successful in continuous black-box optimization. That is, in optimization where no mathematical expression of the optimized function is available, neither an explicit nor implicit one, and it is necessary to empirically evaluate the fitness functions through series of measurements or simulations.

Considering real-world applications, the evaluation of a black-box function can be very time-consuming or expensive. Taking into account this property, the optimization method should evaluate as small amount of points as possible and still reach the target distance to the function optimal value.

The *Covariance Matrix Adaptation Evolution Strategy* (CMA-ES) [4] is considered to be the state-of-the-art optimization algorithm for continuous black-box optimization. On the other hand, the CMA-ES can consume many evaluations to find the optimum of the expensive fitness function. This property resulted in the development of several surrogate-assisted versions of the CMA-ES (an overview can be found in [11]), where a part of evaluations is performed by a regression surrogate model instead of the original fitness function.

The *local meta-model CMA-ES* (lmm-CMA-ES), proposed in [7] and later improved in [1], builds a quadratic regression model for each point using a set of points already evaluated by the fitness function. The convergence of the algorithm is speeded-up by using a control of changes in population ranking after the fraction of the offspring is evaluated by the original fitness.

A different surrogate-assisted approach, utilizing an ordinal SVM to estimate the ranking of the fitness function values, called ^{s*}ACM-ES, has been introduced in [8] and later improved in BIPOP-^{s*}ACM-ES-k [9] to be more robust against premature convergence to local optima. The parameters of the SVM surrogate model are themselves optimized using the CMA-ES algorithm.

In 2016, the *Doubly Trained Surrogate CMA-ES* (DTS-CMA-ES) algorithm, using the ability of Gaussian processes to provide the distribution of predicted points, was introduced in [10]. The algorithm employs uncertainty criteria to choose the most promising points to be evaluated by the original fitness.

Results obtained with the three above-mentioned surrogate-assisted algorithms on noiseless functions [11] suggest that on some fitness functions (e. g., *attractive sector* function) the surrogate model happens to suffer from a loss of accuracy. Whereas the first of these algorithms controls the number of points evaluated by the original fitness function to prevent the model from misleading the search, the DTS-CMA-ES has the amount of evaluated points fixed. Therefore, some control of the amount of points evaluated by the original fitness in each generation could speed-up the DTS-CMA-ES convergence.

This paper extends the original DTS-CMA-ES with an online adaptation of the number of the points evaluated by the original fitness. This extended version of DTS-

Algorithm 1 DTS-CMA-ES [10]

Input: λ (population-size), y_{target} (target value), f (original fitness function), α (ratio of original-evaluated points), $\mathcal{C}$ (uncertainty criterion)

$\sigma, \mathbf{m}, \mathbf{C} \leftarrow$ CMA-ES initialize
$\mathcal{A} \leftarrow \varnothing$ {*archive initialization*}
while minimal y_k from $\mathcal{A} > y_{\text{target}}$ **do**
  $\{\mathbf{x}_k\}_{k=1}^{\lambda} \sim \mathcal{N}(\mathbf{m}, \sigma^2 \mathbf{C})$ {*CMA-ES sampling*}
  $f_{\mathcal{M}1} \leftarrow \text{trainModel}(\mathcal{A}, \sigma, \mathbf{m}, \mathbf{C})$ {*model training*}
  $(\hat{\mathbf{y}}, \mathbf{s}^2) \leftarrow f_{\mathcal{M}1}([\mathbf{x}_1, \ldots, \mathbf{x}_\lambda])$ {*model evaluation*}
  $\mathbf{X}_{\text{orig}} \leftarrow$ select $\lceil \alpha\lambda \rceil$ best points accord. to $\mathcal{C}(\hat{\mathbf{y}}, \mathbf{s}^2)$
  $\mathbf{y}_{\text{orig}} \leftarrow f(\mathbf{X}_{\text{orig}})$ {*original fitness evaluation*}
  $\mathcal{A} = \mathcal{A} \cup \{(\mathbf{X}_{\text{orig}}, \mathbf{y}_{\text{orig}})\}$ {*archive update*}
  $f_{\mathcal{M}2} \leftarrow \text{trainModel}(\mathcal{A}, \sigma, \mathbf{m}, \mathbf{C})$ {*model retrain*}
  $\mathbf{y} \leftarrow f_{\mathcal{M}2}([\mathbf{x}_1, \ldots, \mathbf{x}_\lambda])$ {2^{nd} *model prediction*}
  $(\mathbf{y})_k \leftarrow y_{\text{orig},i}$ for all original-evaluated $y_{\text{orig},i} \in \mathbf{y}_{\text{orig}}$
  $\sigma, \mathbf{m}, \mathbf{C} \leftarrow$ CMA-ES update
end while
$\mathbf{x}_{\text{res}} \leftarrow \mathbf{x}_k$ from $\mathcal{A}$ where y_k is minimal

Output: $\mathbf{x}_{\text{res}}$ (point with minimal y)

Algorithm 2 Adaptive DTS-CMA-ES

Input: λ (population-size), y_{target} (target value), f (original fitness function), β (update rate), α^0, $\alpha_{\min}$, $\alpha_{\max}$ (initial, minimal, and maximal ratio of original-evaluated points), $\mathcal{C}$ (uncertainty criterion), $\mathcal{E}^{(0)}$, $\mathcal{E}_{\min}$, $\mathcal{E}_{\max}$ (initial, minimal, and maximal error)

1: $\sigma, \mathbf{m}, \mathbf{C}, g \leftarrow$ CMA-ES initialize
2: $\mathcal{A} \leftarrow \varnothing$ {*archive initialization*}
3: **while** minimal y_k from $\mathcal{A} > y_{\text{target}}$ **do**
4:   $\{\mathbf{x}_k\}_{k=1}^{\lambda} \sim \mathcal{N}(\mathbf{m}, \sigma^2 \mathbf{C})$ {*CMA-ES sampling*}
5:   $f_{\mathcal{M}1} \leftarrow \text{trainModel}(\mathcal{A}, \sigma, \mathbf{m}, \mathbf{C})$ {*model training*}
6:   $(\hat{\mathbf{y}}, \mathbf{s}^2) \leftarrow f_{\mathcal{M}1}([\mathbf{x}_1, \ldots, \mathbf{x}_\lambda])$ {*model evaluation*}
7:   $\mathbf{X}_{\text{orig}} \leftarrow$ select $\lceil \alpha\lambda \rceil$ best points accord. to $\mathcal{C}(\hat{\mathbf{y}}, \mathbf{s}^2)$
8:   $\mathbf{y}_{\text{orig}} \leftarrow f(\mathbf{X}_{\text{orig}})$ {*fitness evaluation*}
9:   $\mathcal{A} = \mathcal{A} \cup \{(\mathbf{X}_{\text{orig}}, \mathbf{y}_{\text{orig}})\}$ {*archive update*}
10:   $f_{\mathcal{M}2} \leftarrow \text{trainModel}(\mathcal{A}, \sigma, \mathbf{m}, \mathbf{C})$ {*model retrain*}
11:   $\mathbf{y} \leftarrow f_{\mathcal{M}2}([\mathbf{x}_1, \ldots, \mathbf{x}_\lambda])$ {2^{nd} *model prediction*}
12:   $(\mathbf{y})_k \leftarrow y_{\text{orig},i}$ for all original-evaluated $y_{\text{orig},i} \in \mathbf{y}_{\text{orig}}$
13:   $\mathcal{E}^{\text{RDE}} \leftarrow \text{RDE}_\mu(\hat{\mathbf{y}}, \mathbf{y})$ {*model's error estimation*}
14:   $\mathcal{E}^{(g)} \leftarrow (1-\beta)\mathcal{E}^{(g-1)} + \beta\mathcal{E}^{\text{RDE}}$ {*exponen. smooth*}
15:   $\alpha \leftarrow$ update using linear transfer function in Eq.(2)
16:   $\sigma, \mathbf{m}, \mathbf{C}, g \leftarrow$ CMA-ES update
17: **end while**
18: $\mathbf{x}_{\text{res}} \leftarrow \mathbf{x}_k$ from $\mathcal{A}$ where y_k is minimal

Output: $\mathbf{x}_{\text{res}}$ (point with minimal y)

CMA-ES is compared with the original version as well as with the other two above mentioned surrogate models on the noiseless part of the *Comparing-Continuous-Optimisers (COCO) platform* [5, 6] in the expensive scenario and compares it to the original CMA-ES, lmm-CMA-ES, ^{s*}ACM-ES, and original DTS-CMA-ES. Section 2 describes the original DTS-CMA-ES in more detail. Section 3 defines the adaptivity employed to improve the original DTS. Section 4 contains the experimental part. Section 5 summarizes the results and concludes the paper.

2 Doubly Trained Surrogate CMA-ES

The DTS-CMA-ES, introduced in [10], is outlined in Algorithm 1. The algorithm utilizes the ability of GP to estimate the whole probability distribution of fitness to select individuals out of the current population using some uncertainty criterion. The selected individuals are subsequently reevaluated with the original fitness and incorporated into the set of points utilized for retraining the GP model. The CMA-ES strategy parameters (σ, $\mathbf{m}$, $\mathbf{C}$, etc.) are calculated using the original CMA-ES algorithm.

3 Adaptivity for the DTS-CMA-ES

In this section, we propose a simple adaptation mechanism for the DTS-CMA-ES. In DTS-CMA-ES, the number of points evaluated by the original fitness function in one generation is controlled by the ratio α. The higher values of α, the more points are evaluated by the original fitness. As a consequence, more training points are available for the surrogate model around the current CMA-ES mean $\mathbf{m}$. In addition, the CMA-ES is less misled by a smaller amount of points evaluated by the model. On the other hand, the lower values of α imply less evaluations by the original fitness and possibly a faster convergence of the algorithm. The ratio α can be controlled according to the surrogate model precision. Taking into account that the CMA-ES is dependent only on the ordering of the μ best individuals from the current population, we suggest to use the Ranking Difference Error described in the following paragraph.

The *Ranking Difference Error* (RDE_μ), is a normalized version of the error measure used by Kern in [7]. It is the sum of differences of rankings of the μ best points in the population of size λ, normalized by the maximal possible such error for the respective μ, λ ($\rho(i)$ and $\hat{\rho}(i)$ are the ranks of the i-th element in vectors $\mathbf{y}$ and $\hat{\mathbf{y}}$ respectively, where $\mathbf{y}$'s ranking is expected to be more precise)

$$\text{RDE}_\mu(\hat{\mathbf{y}}, \mathbf{y}) = \frac{\sum_{i:\rho(i)\le\mu} |\hat{\rho}(i) - \rho(i)|}{\max_{\pi \in \text{Permutations of }(1,\ldots,\lambda)} \sum_{i:\pi(i)\le\mu} |i - \pi(i)|}. \tag{1}$$

The adaptive DTS-CMA-ES (aDTS-CMA-ES), depicted in Algorithm 2, differs from the original DTS-CMA-ES in several additional steps (lines 13–15) processed after the surrogate model $f_{\mathcal{M}2}$ is retrained using the new original-evaluated points from the current generation (line 10). First, the quality of the model is estimated using the $\text{RDE}_\mu(\hat{\mathbf{y}}, \mathbf{y})$ measured between the first model's prediction $\hat{\mathbf{y}}$ and the vector $\mathbf{y}$ which is composed of the available original fitness values (from $\mathbf{y}_{\text{orig}}$) and the retrained model's predictions for the points which are not original-evaluated. Due to noisy observation of the model's error

$\mathcal{E}^{\mathrm{RDE}}$, we have employed exponential smoothing of the measured error using the update rate β (line 14). As the next step (line 15), α is calculated via linear transfer function of $\mathcal{E}^{(g)}$

$$\alpha = \begin{cases} \alpha_{\min} & \mathcal{E}^{(g)} \le \mathcal{E}_{\min} \\ \alpha_{\min} + \frac{\mathcal{E}^{(g)} - \mathcal{E}_{\min}}{\mathcal{E}_{\max} - \mathcal{E}_{\min}} (\alpha_{\max} - \alpha_{\min}) & \mathcal{E}^{(g)} \in (\mathcal{E}_{\min}, \mathcal{E}_{\max}) \\ \alpha_{\max} & \mathcal{E}^{(g)} \ge \mathcal{E}_{\max} \end{cases}, \tag{2}$$

where $\mathcal{E}_{\min}$ and $\mathcal{E}_{\max}$ are lower and upper bounds for saturation to the values of $\alpha_{\min}$ and $\alpha_{\max}$ respectively.

Having analyzed the RDE_μ error measures on the COCO/BBOB testbed, we observed that the measured RDE error $\mathcal{E}^{(g)}$ depends on the ratio α and the dimension D:

$$\mathcal{E}_{\min} = f^{\mathcal{E}}_{\min}(\alpha, D), \quad \mathcal{E}_{\max} = f^{\mathcal{E}}_{\max}(\alpha, D). \tag{3}$$

Especially dependence on α is not surprising: from the definition of RDE_μ follows that the more reevaluated points, the higher number of summands in nominator of (1) and hence the higher RDE_μ value. Due to mutual dependence of the parameters $\mathcal{E}$ and α, the calculation of α in each generation is performed in a cycle until convergence of α:

(1) calculate error thresholds $\mathcal{E}_{\min}$, $\mathcal{E}_{\max}$ using the last used ratio α – either from the previous iteration, or from the previous generation (see equation (3))

(2) calculate new ratio α using newly calculated $\mathcal{E}_{\min}$, $\mathcal{E}_{\max}$ (see equation (2))

In our implementation, the functions $f^{\mathcal{E}}_{\min}$ and $f^{\mathcal{E}}_{\min}$ are results of multiple linear regression – see section 4.1 for the details of these linear models. The remaining parts of the algorithm are similar to the original DTS-CMA-ES.

4 Experimental Evaluation

In this section, we compared the performances of the aDTS-CMA-ES to the original DTS-CMA-ES [10], the CMA-ES [4], and two other surrogate-assisted versions of the CMA-ES, the lmm-CMA-ES [1, 7] and the ^{s*}ACM-ES [9], on the noiseless part of the COCO/BBOB framework [5, 6].

4.1 Experimental Setup

The considered algorithms were evaluated using the 24 noiseless COCO/BBOB single-objective benchmarks [5, 6] in dimensions $D = 2, 3, 5$ and 10 on 15 different instances per function. The functions were divided into three groups according to the difficulty of their modeling with a GP model, where two groups were used for tuning aDTS-CMA-ES parameters and the remaining group was utilized to test the results of that tuning. The method of dividing the functions into those groups will be described below in connection with the aDTS-CMA-ES settings. The experiment stopping criteria were reaching either the maximum budget of 250 function evaluations per dimension (FE/D), or reaching the target distance from the function optimum $\Delta f_T = 10^{-8}$. The following paragraphs summarize the parameters of the compared algorithms.

The original CMA-ES was tested in its IPOP-CMA-ES version (Matlab code v. 3.61) with the following settings: the number of restarts = 4, IncPopSize = 2, $\sigma_{\text{start}} = \frac{8}{3}$, $\lambda = 4 + \lfloor 3 \log D \rfloor$. The remaining settings were left default.

The lmm-CMA-ES was employed in its improved version published in [1]. The results have been downloaded from the COCO/BBOB results data archive [1] in its GECCO 2013 settings.

We have used the bi-population version of the ^{s*}ACM-ES, the BIPOP-^{s*}ACM-ES-k [9]. Similarly to the lmm-CMA-ES, the algorithm results have also been downloaded from the COCO/BBOB results data archive[2].

The original DTS-CMA-ES was tested using the overall best settings from [10]: the prediction variance of Gaussian process model as the uncertainty criterion, the population size $\lambda = 8 + \lfloor 6 \log D \rfloor$, and the ratio of points evaluated by the original fitness $\alpha = 0.05$. The results of the DTS-CMA-ES are slightly different from previously published results [10, 11] due to a correction of a bug in the original version which was affecting the selection of points to be evaluated by the original fitness using an uncertainty criterion.

The aDTS-CMA-ES was tested with multiple settings of parameters. First, the linear regression models of lower and upper bounds for the error measure $\mathcal{E}_{\min}$, $\mathcal{E}_{\max}$ were identified via measuring RDE_μ on datasets from DTS-CMA-ES runs on the COCO/BBOB benchmarks.

As a first step, we figured out six BBOB functions which are the *easiest* (E) and six which are the *hardest* (H) to regress by our Gaussian process model based on the RDE_μ measured on 1250 *independent testsets* per function in each dimension: 10 sets of λ points in each of 25 equidistantly selected generations from the DTS-CMA-ES runs on the first 5 instances, see Table 1 for these sets of functions and their respective errors. The functions which were not identified as E or H form the *test* function set.

Using the same 25 DTS-CMA-ES "snapshots" on each of 5 instances, we calculated medians (Q_2) and the third quartiles (Q_3) of *measured* RDE_μ on populations from both groups of functions (E) and (H), where we used five different proportions of original-evaluated points $\alpha = \{0.04, 0.25, 0.5, 0.75, 1.00\}$ which were available for retrained models and thus also for measuring models' errors $\mathcal{E}^{(g)}$. These quartiles were regressed by multiple linear regression models using stepwise regression from a full quadratic model of the ratio α and dimension D or its logarithm $\log(D)$ (decision whether to use $\log(D)$ or D

[1] http://coco.gforge.inria.fr/data-archive/2013/lmm-CMA-ES_auger_noiseless.tgz

[2] http://coco.gforge.inria.fr/data-archive/2013/BIPOP-saACM-k_loshchilov_noiseless.tgz

was according to the RMSE of the final stepwise models); the stepwise regression was removing terms with the highest p-value > 0.05. The coefficients $\mathcal{E}^{Q2}_{\min}$ and $\mathcal{E}^{Q3}_{\min}$ of the lower thresholds were estimated on the data from (E) and the coefficients $\mathcal{E}^{Q2}_{\max}$ and $\mathcal{E}^{Q3}_{\max}$ of the higher thresholds on the data from (H), which resulted in the following models:

$$\begin{array}{lllllll} \mathcal{E}^{Q2}_{\min}(\alpha,D) & = & (1 & \log(D) & \alpha & \alpha\log(D) & \alpha^2) \cdot b_1 \\ \mathcal{E}^{Q3}_{\min}(\alpha,D) & = & (1 & D & \alpha & \alpha D & \alpha^2) \cdot b_2 \\ \mathcal{E}^{Q2}_{\max}(\alpha,D) & = & (1 & D & \alpha & \alpha D & \alpha^2) \cdot b_3 \\ \mathcal{E}^{Q3}_{\max}(\alpha,D) & = & (1 & \log(D) & \alpha & \alpha\log(D) & \alpha^2) \cdot b_4 \end{array}$$

where

$$b_1 = \begin{pmatrix} 0.11 \\ -0.0092 \\ -0.13 \\ 0.044 \\ 0.14 \end{pmatrix} \quad b_2 = \begin{pmatrix} 0.17 \\ -0.00067 \\ -0.095 \\ 0.0087 \\ 0.15 \end{pmatrix} \quad b_3 = \begin{pmatrix} 0.18 \\ -0.0027 \\ 0.44 \\ 0.0032 \\ -0.14 \end{pmatrix} \quad b_4 = \begin{pmatrix} 0.35 \\ -0.047 \\ 0.44 \\ 0.044 \\ -0.19 \end{pmatrix}.$$

For the remaining investigations, three different values of exponential smoothing update rate were used for comparison $\beta = \{0.3, 0.4, 0.5\}$. The minimal and maximal values of α were set to $\alpha_{\min} = 0.04$ and $\alpha_{\max} = 1.0$ because lower α values than 0.04 would yield to less than one original-evaluated point per generation, and the aDTS-CMA-ES has to be able to spend the whole populations for the original evaluations in order to work well on functions where GP model is poor (e. g., on f_6 *Attractive sector*). The initial error and original ratio values were set to $\mathcal{E}^{(0)} = 0.05$ and $\alpha^0 = 0.05$. The rest of aDTS-CMA-ES parameters were left the same as in the original DTS-CMA-ES settings.

4.2 Results

The results in Figures 1, 2, and 3 and in Table 3 show the effect of adaptivity implemented in the DTS-CMA-ES. The graphs in Figures 1, 2 and 3 depict the scaled logarithm $\Delta^{\log}_f$ of the median Δ^{med}_f of minimal distances from the function optimum over runs on 15 independent instances as a function of FE/D. The scaled logarithms of Δ^{med}_f are calculated as

$$\Delta^{\log}_f = \frac{\log\Delta^{\text{med}}_f - \Delta^{\text{MIN}}_f}{\Delta^{\text{MAX}}_f - \Delta^{\text{MIN}}_f} \log_{10}\left(1/10^{-8}\right) + \log_{10} 10^{-8}$$

where Δ^{MIN}_f (Δ^{MAX}_f) is the minimum (maximum) $\log\Delta^{\text{med}}_f$ found among all the compared algorithms for the particular function f and dimension D between 0 and 250 FE/D. Such scaling enables the aggregation of $\Delta^{\log}_f$ graphs across arbitrary number of functions and dimensions (see Figure 3). The values are scaled to the $[-8, 0]$ interval, where -8 corresponds to the minimal and 0 to the maximal distance. This visualization has a better ability to distinguish the differences in the convergence of tested algorithms more than the default visualization used by the COCO/BBOB platform and that is why it was used in this article.

We have tested the statistical significance of differences in algorithms' performance on 12 COCO/BBOB *test* functions in 10D for separately two evaluation budgets using the Iman and Davenport's improvement of the Friedman test [2]. Let $\#\text{FE}_\text{T}$ be the smallest number of function evaluations on which at least one algorithm reached the target, i. e., satisfied $\Delta^{\text{med}}_f \leq \Delta f_T$, or $\#\text{FE}_\text{T} = 250D$ if no algorithm reached the target within 250D evaluations. The algorithms are ranked on each COCO/BBOB *test* function with respect to Δ^{med}_f at a given budget of function evaluations. The null hypothesis of equal performance of all algorithms is rejected at a higher function evaluation budget $\#\text{FEs} = \#\text{FE}_\text{T}$ $(p < 10^{-3})$, as well as at a lower budget $\#\text{FEs} = \frac{\#\text{FE}_\text{T}}{3}$ $(p < 10^{-3})$.

We test pairwise differences in performance utilizing the post-hoc Friedman test [3] with the Bergmann-Hommel correction controlling the family-wise error in cases when the null hypothesis of equal algorithms' performance was rejected. To illustrate algorithms' differences, the numbers of *test* functions at which one algorithm achieved a higher rank than the other are reported in Table 3. The table also contains the pairwise statistical significances.

We have compared the performances of aDTS-CMA-ES using twelve settings differing in $\mathcal{E}_{\min}$, $\mathcal{E}_{\max}$, and β. Table 2 illustrates the counts of the 1st ranks of the compared settings according to the lowest achieved Δ^{med}_f for 25, 50, 100, and 200 FE/D respectively. The counts are summed across the testing sets of benchmark functions in each individual dimension.

Although the algorithm is rather robust to exact setting of smoothing update rate, we have found that the lower the β, the better the performance is usually observed (see Table 2), and thus the following experiments use the rate $\beta = 0.3$.

When comparing the convergence rate, the performance of aDTS-CMA-ES with $\mathcal{E}^{Q2}_{\min}$ is noticeable lower especially on Rosenbrock's functions (f_8, f_9) and Different powers f_{14} where the RDE_μ error often exceeds the lower error threshold even if a lower number of original-evaluated points would be sufficient for higher speedup of the CMA-ES. The adaptive control, on the other hand, helps especially on the Attractive sector f_6, which has the optimum in a point without continues derivatives and is therefore hard-to-regress by GPs, or on Shaffers' functions f_{17}, f_{18} where the aDTS-CMA-ES is probably able to adapt to multimodal neighbourhood around function's optimum and performs best of all the compared algorithms. Within the budget of 250 FE/D, the aDTS-CMA-ES (especially with $\mathcal{E}^{Q2}_{\min}$) is also able to find one of the best fitness value on regularly multimodal Rastrigin functions f_3, f_4 or f_{15} where the GP model apparently does not prevent the original CMA-ES from exploiting the global structure of a function.

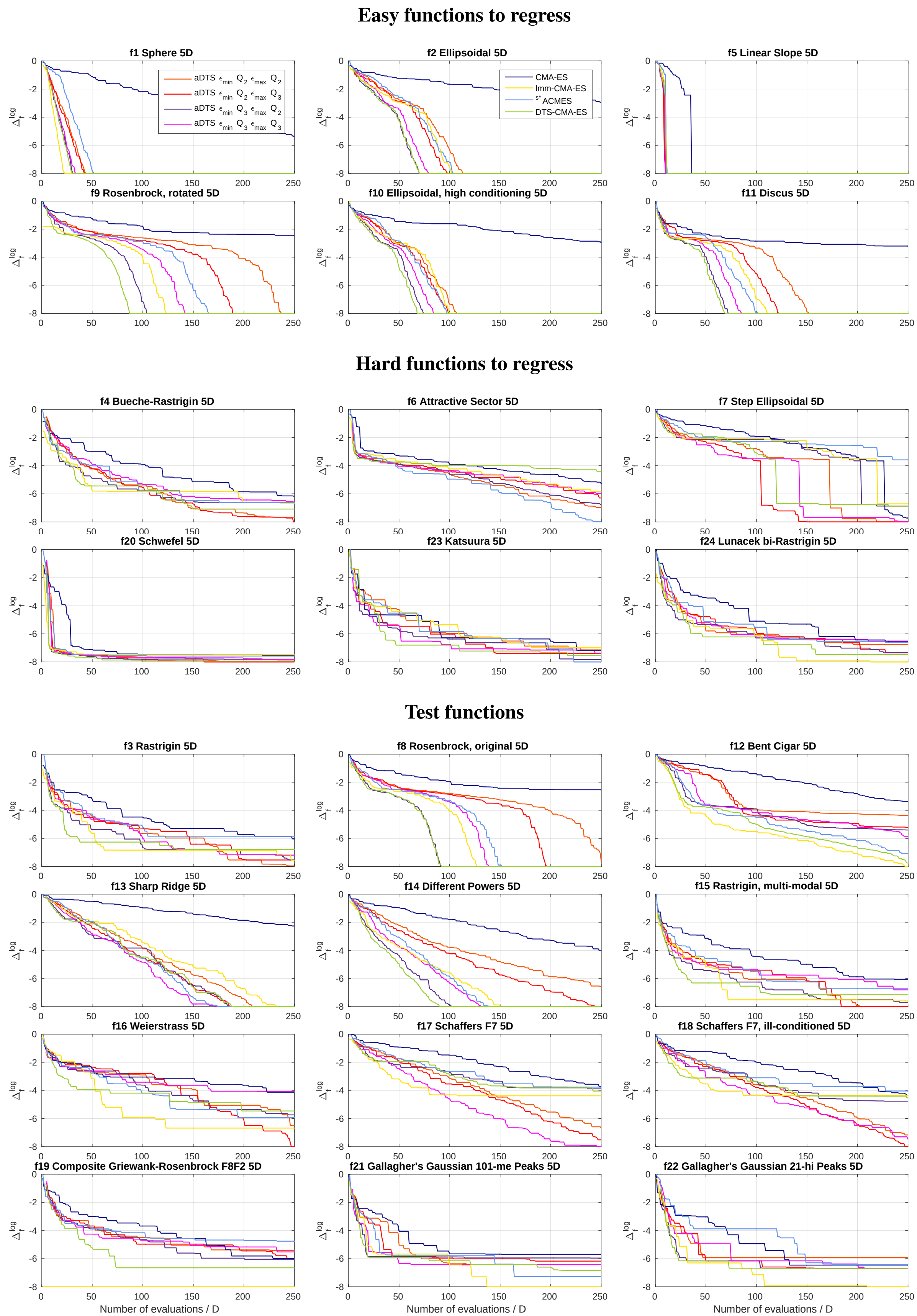

Figure 1: Algorithm comparison on 24 COCO/BBOB noiseless functions in 5D. ε_{min}, ε_{max}: minimal and maximal error, $Q2$, $Q3$: median and third quartile.

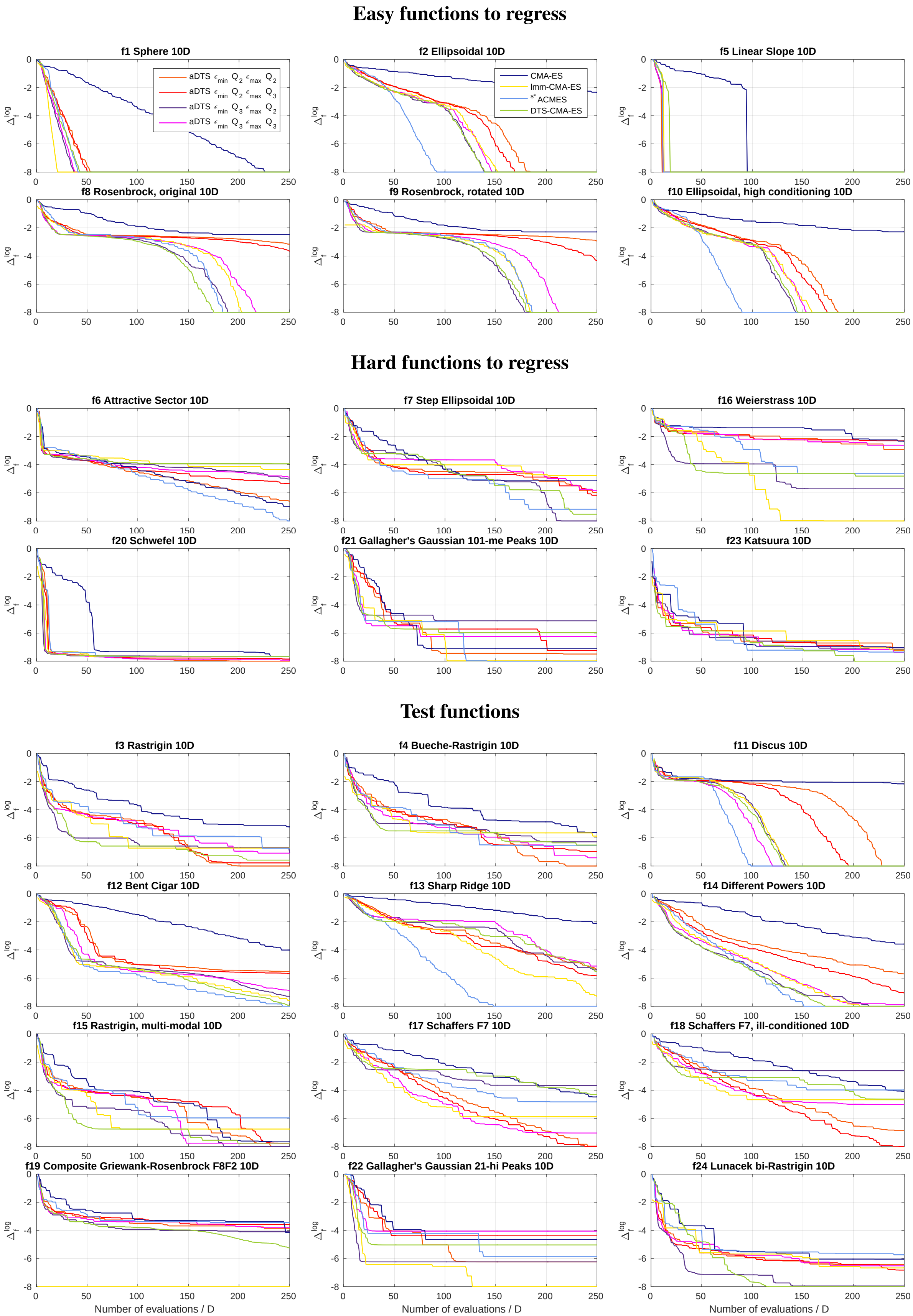

Figure 2: Algorithm comparison on 24 COCO/BBOB noiseless functions in $10D$. $\varepsilon_{\min}$, $\varepsilon_{\max}$: minimal and maximal error, $Q2$, $Q3$: median and third quartile.

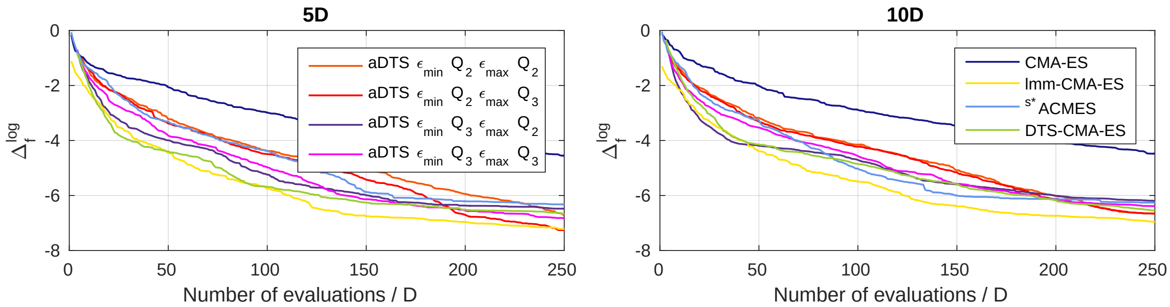

Figure 3: Algorithm comparison using averaged $\Delta_f^{\log}$ values on 12 *test* functions from the COCO/BBOB testbed in 5D and 10D. $\varepsilon_{\min}$, $\varepsilon_{\max}$: minimal and maximal error, $Q2$, $Q3$: median and third quartile.

Table 1: The *easiest* (1.–6.) and the *hardest* (19.–24.) to regress six COCO/BBOB functions by the Gaussian process used in the DTS-CMA-ES (columns f) according to the corresponding medians of RDE_μ error measured in 25 generations from 5 instances on independent testsets of size $\lambda = 8 + \lfloor 6 \log D \rfloor$ using $\mu = \lambda/2$.

	2D		3D		5D		10D		20D	
	f	RDE_μ	f	RDE_μ	f	RDE_μ	f	RDE_μ	f	RDE_μ
1.	5	0.00	5	0.00	5	0.00	5	0.04	5	0.04
2.	1	0.08	1	0.11	1	0.16	1	0.18	1	0.08
3.	2	0.10	2	0.13	10	0.18	10	0.26	24	0.18
4.	10	0.10	10	0.14	2	0.21	8	0.27	15	0.19
5.	11	0.10	9	0.16	11	0.23	2	0.27	19	0.20
6.	8	0.14	8	0.18	9	0.24	9	0.29	3	0.21
19.	18	0.46	23	0.43	24	0.41	21	0.40	18	0.38
20.	20	0.52	15	0.43	4	0.44	16	0.41	23	0.47
21.	24	0.53	24	0.49	23	0.45	23	0.47	6	0.48
22.	6	0.54	20	0.51	6	0.51	6	0.50	21	0.51
23.	7	0.54	6	0.52	20	0.54	20	0.54	20	0.52
24.	19	0.54	7	0.54	7	0.56	7	0.57	7	0.56

Table 2: Counts of the 1st ranks from 12 benchmark *test* functions from the BBOB/COCO testbed according to the lowest achieved Δ_f^{med} for different $\text{FE}/D = \{25, 50, 100, 200\}$ and dimensions $D = \{2, 3, 5, 10\}$. Ties of the 1st ranks are counted for all respective algorithms. The ties often occure when $\Delta f_T = 10^{-8}$ is reached (mostly on f_1 and f_5).

	2D				3D				5D				10D				$\sum$			
FE/D	25	50	100	200	25	50	100	200	25	50	100	200	25	50	100	200	25	50	100	200
$\varepsilon_{\min}^{Q2}, \varepsilon_{\max}^{Q2}, \beta = 0.3$	1	0	0	2	1	2	2	3	0	0	0	0	0	0	0	1	2	2	2	6
$\varepsilon_{\min}^{Q2}, \varepsilon_{\max}^{Q2}, \beta = 0.4$	0	0	0	2	1	0	0	3	0	0	0	0	0	0	0	0	1	0	0	5
$\varepsilon_{\min}^{Q2}, \varepsilon_{\max}^{Q2}, \beta = 0.5$	2	0	0	3	0	0	0	3	0	0	0	0	0	0	0	1	2	0	0	7
$\varepsilon_{\min}^{Q2}, \varepsilon_{\max}^{Q3}, \beta = 0.3$	2	0	1	4	1	0	0	3	0	0	2	2	0	0	0	2	3	0	3	11
$\varepsilon_{\min}^{Q2}, \varepsilon_{\max}^{Q3}, \beta = 0.4$	0	0	1	4	0	0	0	4	0	1	2	3	0	0	**3**	**4**	0	1	6	15
$\varepsilon_{\min}^{Q2}, \varepsilon_{\max}^{Q3}, \beta = 0.5$	1	0	0	3	0	0	0	4	0	0	0	3	0	0	0	1	1	0	0	11
$\varepsilon_{\min}^{Q3}, \varepsilon_{\max}^{Q2}, \beta = 0.3$	1	0	**3**	4	**4**	**5**	**5**	**5**	**5**	3	**3**	**5**	**7**	**6**	**3**	**4**	**17**	**14**	**14**	**18**
$\varepsilon_{\min}^{Q3}, \varepsilon_{\max}^{Q2}, \beta = 0.4$	1	**4**	1	4	1	0	1	4	**5**	**4**	0	4	1	2	**3**	2	8	10	5	14
$\varepsilon_{\min}^{Q3}, \varepsilon_{\max}^{Q2}, \beta = 0.5$	1	3	1	**5**	0	2	2	3	1	0	1	2	3	0	0	1	5	5	4	11
$\varepsilon_{\min}^{Q3}, \varepsilon_{\max}^{Q3}, \beta = 0.3$	0	2	2	4	1	3	3	3	0	3	**3**	4	0	1	2	1	1	9	10	12
$\varepsilon_{\min}^{Q3}, \varepsilon_{\max}^{Q3}, \beta = 0.4$	0	2	**3**	4	3	0	3	4	1	1	0	4	0	2	0	1	4	5	6	13
$\varepsilon_{\min}^{Q3}, \varepsilon_{\max}^{Q3}, \beta = 0.5$	**3**	1	2	3	0	0	1	3	0	0	1	2	1	1	1	2	4	2	5	10

Table 3: A pairwise comparison of the algorithms on 12 *test* functions in 10D over the COCO/BBOB for different evaluation budgets. The number of wins of i-th algorithm against j-th algorithm over all benchmark functions is given in i-th row and j-th column. The asterisk marks the row algorithm being significantly better than the column algorithm according to the Friedman post-hoc test with the Bergmann-Hommel correction at family-wise significance level $\alpha = 0.05$.

10D	$\varepsilon^{Q2}_{\min}$, $\varepsilon^{Q2}_{\max}$		$\varepsilon^{Q2}_{\min}$, $\varepsilon^{Q3}_{\max}$		$\varepsilon^{Q3}_{\min}$, $\varepsilon^{Q2}_{\max}$		$\varepsilon^{Q3}_{\min}$, $\varepsilon^{Q3}_{\max}$		CMA-ES		lmm-CMA-ES		^{s*}ACM-ES		DTS-CMA-ES	
$^{\#\mathrm{FEs}}/_{\#\mathrm{FE_T}}$	1/3	1	1/3	1	1/3	1	1/3	1	1/3	1	1/3	1	1/3	1	1/3	1
$\varepsilon^{Q2}_{\min}$, $\varepsilon^{Q2}_{\max}$	—	—	6	5	4	6	4	8	12	11	2	5	6	8	3	7
$\varepsilon^{Q2}_{\min}$, $\varepsilon^{Q3}_{\max}$	6	7	—	—	3	5	5	6	11	10	2	6	7	7	2	5
$\varepsilon^{Q3}_{\min}$, $\varepsilon^{Q2}_{\max}$	8	6	9	7	—	—	7	7	12*	9	4	5	8	5	3	2
$\varepsilon^{Q3}_{\min}$, $\varepsilon^{Q3}_{\max}$	8	4	7	6	5	5	—	—	11*	10	4	6	9	7	4	4
CMA-ES	0	1	1	2	0	3	1	2	—	—	1	1	2	4	0	1
lmm-CMA-ES	10	7	10	6	8	7	8	6	11*	11	—	—	10	6	7	5
^{s*}ACM-ES	6	4	5	5	4	7	3	5	10	8	2	6	—	—	3	7
DTS-CMA-ES	9	5	10	7	9	10	8	8	12*	11*	5	7	9	5	—	—

5 Conclusions & Future work

In this paper, we have presented a work-in-progress on adaptive version of the surrogate-assisted optimization algorithm DTS-CMA-ES. The online adjustment of the ratio between the original- and model-evaluated points according to the error of the surrogate model is investigated. The new adaptive version of the algorithm employs RDE_μ between the fitness of current population predicted using the first-trained model and the retrained model.

Results of parameter tunning show that lower values of the exponential smoothing rate β provide better results. On the other hand, different combinations of slower and more rapid update behaviours bring better CMA-ES speedup for different kinds of functions, and choice of this parameter could depend on the experimenter's domain knowledge. We found that the adaptive approach speeds up the CMA-ES more than three other surrogate CMA-ES algorithms, namely DTS-CMA-ES, ^{s*}ACM-ES, and lmm-CMA-ES, on several functions after roughly 150 FE$/D$.

The adaptivity of the DTS-CMA-ES is still, to a certain extent, work in progress. A future perspective of improving aDTS-CMA-ES is to additionally investigate different types and properties of adaptive control of the number of points evaluated by the original fitness in each generation. Another conceivable direction of future research can be found in online switching between different types of surrogate models suitable for the aDTS-CMA-ES.

Acknowledgements

The reported research was supported by the Czech Science Foundation grant No. 17-01251, by the Grant Agency of the Czech Technical University in Prague with its grant No. SGS17/193/OHK4/3T/14, and by the project Nr. LO1611 with a financial support from the MEYS under the NPU I program. Further, access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the programme "Projects of Large Infrastructure for Research, Development, and Innovations" (LM2010005), is greatly appreciated.

References

[1] A. Auger, D. Brockhoff, and N. Hansen. Benchmarking the local metamodel CMA-ES on the noiseless BBOB'2013 test bed. In *Proceedings of the GECCO '13 Companion*, 2013.

[2] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[3] S. García and F. Herrera. An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research*, 9:2677–2694, 2008.

[4] N. Hansen. The CMA Evolution Strategy: A Comparing Review. In *Towards a New Evolutionary Computation*, number 192, pages 75–102. Springer, 2006.

[5] N. Hansen, A. Auger, S. Finck, and R. Ros. Real-parameter black-box optimization benchmarking 2012: Experimental setup. Technical report, INRIA, 2012.

[6] N. Hansen, S. Finck, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions. Technical Report RR-6829, INRIA, 2009. Updated February 2010.

[7] S. Kern, N. Hansen, and P. Koumoutsakos. Local Meta-models for Optimization Using Evolution Strategies. In *PPSN IX Proceedings*, volume 4193 of *Lecture Notes in Computer Science*, pages 939–948. Springer, 2006.

[8] I. Loshchilov, M. Schoenauer, and M. Sebag. Self-adaptive surrogate-assisted covariance matrix adaptation evolution strategy. In *Proceedings of the GECCO '12*, pages 321–328. ACM, 2012.

[9] I. Loshchilov, M. Schoenauer, and M. Sebag. BI-population CMA-ES Algorithms with Surrogate Models and Line Searches. In *Proceedings of the GECCO '13 Companion*, pages 1177–1184. ACM Press, 2013.

[10] Z. Pitra, L. Bajer, and M. Holeňa. Doubly trained evolution control for the Surrogate CMA-ES. In *PPSN XIV Proceedings*, pages 59–68. Springer, 2016.

[11] Z. Pitra, L. Bajer, J. Repický, and M. Holeňa. Overview of surrogate-model versions of covariance matrix adaptation evolution strategy. In *Proceedings of the GECCO '17 Companion*. ACM, 2017.