Benchmarking Gaussian Processes and Random Forests Surrogate Models on the BBOB Noiseless Testbed

Lukáš Bajer
Institute of Computer Science
Academy of Sciences of the Czech Republic
and
Faculty of Mathematics and Physics
Charles University in Prague
Malostranské nám. 25
Prague, Czech Republic
bajer@cs.cas.cz

Zbyněk Pitra
National Institute of Mental Health, Czech Republic
and
Faculty of Nuclear Sciences and Physical Engineering
Czech Technical University in Prague
Trojanova 13
Prague, Czech Republic
z.pitra@gmail.com

Martin Holeňa
Institute of Computer Science,
Academy of Sciences of the Czech Republic
Pod Vodárenskou věží 2
Prague, Czech Republic
martin@cs.cas.cz

ABSTRACT

Speeding-up black-box optimization algorithms via learning and using a surrogate model is a heavily studied topic. This paper evaluates two different surrogate models: Gaussian processes and random forests which are interconnected with the state-of-the art optimization algorithm CMA-ES. Results on the BBOB testing set show that considerable amount of fitness evaluations can be saved especially during the initial phase of the algorithm's progress.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*global optimization, unconstrained optimization*; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems

Keywords

Benchmarking; Black-box optimization; Surrogate model; Gaussian process; Random forest

1. INTRODUCTION

One of the approaches speeding up optimization of expensive black-box problems is surrogate modelling. This technique tries to decrease the number of expensive fitness evaluations via (usually CPU-intensive) fitting and usage of a regression model of the objective function [10]. The model is trained on the already gathered input–output value pairs $(\mathbf{x}_i, y_i), i = 1, \dots, N$ and is used instead of the original expensive fitness to evaluate some of the points needed by the optimization algorithm.

GECCO '15, July 11 - 15, 2015, Madrid, Spain

DOI: http://dx.doi.org/10.1145/2739482.2768468

Nowadays, the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [5] is considered as a state-of-the-art continuous black-box optimizer. However, it is not the fastest in setups where very few fitness evaluations are possible, so several surrogate-assisted versions have appeared during the last decade. One of the most promising results are reported with Support Vector Machine (SVM) ordinal regression models [9] adopted specially for the CMA-ES.

This paper evaluates two kinds of surrogate models, Gaussian processes (GP) and Random forests (RF), in combination with the CMA-ES. The resulting modified CMA-ES algorithm, temporarily named S-CMA-ES, is benchmarked on the BBOB testing set [6] with two different settings of the algorithm and is compared with the original CMA-ES, too. Compared to the previous version of S-CMA-ES [1], Mahalanobis distance is used for model training and prediction which speeded up the GP-based S-CMA-ES.

2. SURROGATE MODELS FOR CMA-ES

Following Jin's terminology [8], S-CMA-ES uses the *generation-based* evolution control which switches between g_o original- and g_m model-evaluated generations. The individual-based evolution control which samples many candidate points every generation and chooses some of them for the original re-evaluation did not noticeably speed-up the original CMA-ES in our preliminary tests, and is, therefore, not described in this work.

The incorporation of surrogate models within the CMA-ES is rather straightforward. The only point where we have modified the original CMA-ES is the sampling and fitness-evaluating part which is replaced by the Algorithm 1 and, similarly to the original sampling and evaluating code, executed at the beginning of each CMA-ES generation.

As opposed to Loshchilov's ^{s*}aACM-ES [9], S-CMA-ES calls the model training or prediction directly within every generation of the CMA-ES. In order not to circumvent the BBOB/COCO benchmarking toolbox, fitness values in the model-generations are adjusted in such a way that the model-predicted fitness values $\hat{y}_i = f_{\mathcal{M}}(\mathbf{x}_i)$ are never lower (better) than the so-far minimal (best) achieved original fitness value $y_{\text{best}} = \min_{\mathcal{A}} y$ (step 17 in the Algorithm 1).

Algorithm 1 Surrogate CMA-ES Algorithm

Input: g (generation), g_m (number of model generations),
σ, λ, $\mathbf{m}$, $\mathbf{C}$ (CMA-ES internal variables),
r (maximal distance between training points and $\mathbf{m}$),
n_{REQ} (minimal number of points for model training),
n_{MAX} (maximal number of points for model training),
$\mathcal{A}$ (archive), $f_{\mathcal{M}}$ (model), f (original fitness function)

$\mathbf{x}_k \sim \mathcal{N}\left(\mathbf{m}, \sigma^2\mathbf{C}\right) \quad k = 1, \ldots, \lambda \quad$ {*CMA-ES sampling*}
if g is original-evaluated **then**
  $y_k \leftarrow f(\mathbf{x}_k) \quad k = 1, \ldots, \lambda \quad$ {*fitness evaluation*}
  $\mathcal{A} = \mathcal{A} \cup \{(\mathbf{x}_k, y_k)\}_{k=1}^{\lambda}$
  $(\mathbf{X}_{\mathrm{tr}}, \mathbf{y}_{\mathrm{tr}}) \leftarrow \{(\mathbf{x}, y) \in \mathcal{A} \,|\, (\mathbf{m}-\mathbf{x})^{\top}\sigma\mathbf{C}^{-1/2}(\mathbf{m}-\mathbf{x}) \le r\}$
  if $|\mathbf{X}_{\mathrm{tr}}| \ge n_{\mathrm{REQ}}$ **then**
    $(\mathbf{X}_{\mathrm{tr}}, \mathbf{y}_{\mathrm{tr}}) \leftarrow$ choose n_{MAX} points if $|\mathbf{X}_{\mathrm{tr}}| > n_{\mathrm{MAX}}$
    {*transformation to the eigenvector basis:*}
    $\mathbf{X}_{\mathrm{tr}} \leftarrow \{(\sigma\mathbf{C}^{-1/2})^{\top}\mathbf{x}_{\mathrm{tr}}$ for each $\mathbf{x}_{\mathrm{tr}} \in \mathbf{X}_{\mathrm{tr}}\}$
    $f_{\mathcal{M}} \leftarrow \mathrm{trainModel}(\mathbf{X}_{\mathrm{tr}}, \mathbf{y}_{\mathrm{tr}})$
    mark $(g+1)$ as model-evaluated
  else
    mark $(g+1)$ as original-evaluated
  end if
else
  $\mathbf{x}_k \leftarrow (\sigma\mathbf{C}^{-1/2})^{\top}\mathbf{x}_k \quad k = 1, \ldots, \lambda$
  $y_k \leftarrow f_{\mathcal{M}}(\mathbf{x}_k) \quad k = 1, \ldots, \lambda \quad$ {*model evaluation*}
  {*shift y_k values if* $(\min y_k) <$ *best y from $\mathcal{A}$*}
  $y_k = y_k + \max\{0, \ \min_{\mathcal{A}} y - \min y_k\} \quad k = 1, \ldots, \lambda$
  if g_m model generations passed **then**
    mark $(g+1)$ as original-evaluated
  end if
end if

Output: $f_{\mathcal{M}}$, $\mathcal{A}$, $(y_k)_{k=1}^{\lambda}$

2.1 Gaussian Processes

Gaussian process is a probabilistic model based on Gaussian distributions. It is specified by mean and covariance functions $\mu(\mathbf{x})$, $k(\mathbf{x}_1, \mathbf{x}_2, \theta)$, and a relatively small number of hyper-parameters θ which precisely specify the covariance function and which are fitted based on training data $(\mathbf{X}_N, \mathbf{y}_N) = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^D, y_i = f(\mathbf{x}_i)\}_{i=1}^{N}$ (in our case, $f(\mathbf{x})$ is the original fitness function being modelled).

Covariance functions k describe prior assumptions on the shape of the modelled function f; $k(\mathbf{x}_i, \mathbf{x}_j)$ is the covariance between the *function values* at two data points $\mathbf{x}_i$ and $\mathbf{x}_j$, $k(\mathbf{x}_i, \mathbf{x}_j) = cov(f(\mathbf{x}_i), f(\mathbf{x}_j))$, and express the similarity between function values in these two points [11].

The values of the covariance function on all the pairs $(\mathbf{x}_i, \mathbf{x}_j)$ of the N training data form the matrix $\mathbf{K}_N \in \mathbb{R}^{N \times N}$, $\{\mathbf{K}_N\}_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j, \theta)$. The final Gaussian process' covariance matrix $\mathbf{C}_N$ is defined using the noise variance σ_n^2 as $\mathbf{C}_N = \mathbf{K}_N + \sigma_n^2 \mathbf{I}_N$ with $\mathbf{I}_N$ being an identity matrix.

We have used the *Matérn* covariance function with the parameter $\nu = 5/2$ and with $r = (\mathbf{x}_i - \mathbf{x}_j)$

$$K_{\text{Matérn}}^{\nu=5/2}(r) = \sigma_f^2 \left(1 + \frac{\sqrt{5}r}{\ell} + \frac{5r^2}{3\ell^2}\right) \exp\left(-\frac{\sqrt{5}r}{\ell}\right). \quad (1)$$

The parameter ℓ is the characteristic length-scale with which the distance of two considered data points is compared and σ_f^2 is signal variance. Fitting the GP model in our application means tuning the hyper-parameters $\theta = (\sigma_n^2, \ell, \sigma_f^2)$ using the maximum-likelihood method.

2.2 Random Forests

Random forest is actually an ensemble of decision trees [2]. Decision trees have a wide spectrum of forms, utilizations, and properties [4]. So far, we have paid attention only to binary regression trees with real inputs. In such regression trees, each observation $\mathbf{x} = (x_1, x_2, \ldots, x_D) \in \mathbb{R}^D$ passes through a series of binary decisions ($x_i \overset{?}{<} c \in \mathbb{R}$) associated with internal nodes and arrives in the leaf node containing a real-valued constant utilized for the prediction of function values $\mathbf{y}$. Each binary decision determines whether the observation proceeds to the left or right child of the respective internal node.

The tree growing starts with one node (root) and a set of all input data. As the first step, real constants (left and right) for all allowable splits of the input set in all variables of the input space are calculated by averaging function values of training points in respective subsets. As the second step, a split with the minimal mean-squared error (MSE) between the resulting constants and the training points in relevant subsets is chosen, and apropriate child nodes are connected to the root. Those two steps are repeated recursively with the children. The tree growing stops whenever any of the user-defined constraints holds: the tree reaches the allowed maximum of splits, the value of the MSE of the predictors decreases below a specified threshold, or if each node contains at least the defined number of points and any additional split would violate it.

An important aspect of random forests are random differences between individual trees within the ensemble. This increases robustness and improves generalization of the prediction. The forest gains randomness during training by bagging [3].

The overall forest prediction is provided by averaging all tree predictions. This form of prediction means the larger the ensemble the greater the robustness to noise.

3. EXPERIMENTAL SETUP

Four S-CMA-ES algorithms are part of this study: GP1-CMAES, GP5-CMAES, RF1-CMAES and RF5-CMAES. Here, GP/RF denotes the type of the surrogate model, and the number of model-evaluated generations g_m follows. All considered S-CMA-ES versions use the distance $r = 8$ (see algorithm 1), and $g_o = 1$. For the GP model, $K_{\text{Matérn}}^{\nu=5/2}$ covariance function (1) with starting values $(\sigma_n^2, l, \sigma_f^2) = \log(0.01, 2, 0.5)$ has been used. We have tested RF comprising 100 regression trees, each containing at least two training points in each leaf. All the algorithms are based on the same IPOP-CMA-ES (Matlab code v. 3.61) with the following parameters: number of restarts = 4, IncPopSize = 2, $\sigma_{start} = \frac{8}{3}$, $\lambda = 4 + \lfloor 3 \log D \rfloor$. The remainder settings were left default.

4. CPU TIMING

In order to evaluate the CPU timing of the S-CMA-ES algorithms, we have run all the proposed algorithms on the function f_8 until a maximum budget $50D$ evaluations is reached, which was far more than the required 30 seconds. The code was run on an Intel(R) Core(TM)2 Duo CPU E7600 @3.06GHz, 4 GB RAM with 1 processor and 2 cores. Times per function evaluation for dimensions 2, 5, 10, 20 are registered in Table 1 for all four algorithms.

Algorithm	2D	5D	10D	20D
GP1-CMAES	0.2255	0.5713	0.1621	0.2272
GP5-CMAES	0.1870	0.2076	0.1184	0.1645
RF1-CMAES	0.2850	0.2470	0.1155	0.1395
RF5-CMAES	0.9870	0.8532	0.2315	0.1353

Table 1: The time in seconds per function evaluation for dimensions 2, 5, 10, 20 for the S-CMA-ES algorithms.

5. RESULTS

Results from experiments according to [6] on all the 24 noiseless benchmark functions given in [7] are presented in Figures 1, 2 and 3 and in Tables 2 and 3. The **expected running time (ERT)**, used in the figures and tables, depends on a given target function value, $f_t = f_{opt} + \Delta f$, and is computed over all relevant trials as the number of the original function evaluations (FEs) executed during each trial until the best function value reached f_t, summed over all trials and divided by the number of trials that actually reached f_t [6].

The most noticeable speedup of the surrogate-assisted S-CMA-ES can be observed for the "GP5" version of the S-CMA-ES (GP model, $g_m = 5$), especially in case of higher target values f_t, i.e. in earlier parts of the optimization progress. The graphs in Figure 1 reveal that the GP5-CMA-ES obtained the best results on nine functions ($f_2, f_5, f_7, f_{10}, f_{11}, f_{13}, f_{14}, f_{15}, f_{23}$) in at least 3 out of 4 dimensionalities among the four tested S-CMA-ES versions and the original CMA-ES. This fact is also clear from the empirical cumulative distribution functions (ECDF) in Figures 2 and 3 where practically all the central parts (roughly between 10 and 100 FEs/dimension) are dominated by the GP5 model.

However, the dominant algorithms change if we consider lower (tighter) target values or later parts of the optimizations, which can be seen from the ECDF graphs. Here, the GP5-CMA-ES is always outperformed by the GP1-CMA-ES ($g_m = 1$) and often also by other algorithms: by the original CMA-ES and, in $20D$, by the RF1-CMA-ES, too.

Random forest S-CMA-ES outperforms other algorithms only rarely in $5D$, usually in early stages of the optimization (on $f_{12}, f_{13}, f_{16}, f_{17}, f_{22}$ and f_{23}), but it is the best algorithm on 5-dimensional f_{18}. Nevertheless, the RF performance on $20D$ benchmarks is considerably more balanced and the RF models can be here considered more robust than the GP models, which start to suffer from higher dimensions (on f_{12} and f_{20}).

6. CONCLUSION AND FUTURE WORK

We have compared four versions of the recently proposed surrogate-assisted CMA-ES algorithm with regular CMA-ES; Gaussian processes and random forests were used as surrogate models with two different evolution-control settings.

We plan to further investigate both the algorithm and the models. It is clear that the optimization performance suffers from the fixed number of model-evaluated generations for which some kind of self-adaptation will be inquired. Further, better model re-use and re-train will be studied as the models are currently always trained from scratch.

7. ACKNOWLEDGMENTS

This work was supported by the Czech Science Foundation (GAČR) grant P103/13-17187S, by the Grant Agency of the Czech Technical University in Prague with its grant No. SGS14/205/OHK4/3T/14, and by the project "National Institute of Mental Health (NIMH-CZ)", grant number CZ.1.05/2.1.00/03.0078 (and the European Regional Development Fund.). 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.

8. REFERENCES

[1] L. Bajer, Z. Pitra, and M. Holeňa. Investigation of gaussian processes and random forests as surrogate models for evolutionary black-box optimization. In *Proceedings of the 17th GECCO*, Madrid, 2015 (to appear). ACM, New York.

[2] L. Breiman. *Classification and regression trees.* Chapman & Hall/CRC, 1984.

[3] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[4] A. Criminisi, J. Shotton, and E. Konukoglu. Decision forests for classification, regression, density estimation, manifold learning and semi-supervised learning. Technical Report MSR-TR-2011-114, Microsoft Research, 2011.

[5] N. Hansen. The CMA Evolution Strategy: A Comparing Review. In J. A. Lozano, P. Larrañaga, I. Inza, and E. Bengoetxea, editors, *Towards a New Evolutionary Computation*, number 192 in Studies in Fuzziness and Soft Computing, pages 75–102. Springer Berlin Heidelberg, Jan. 2006.

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

[7] 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.

[8] Y. Jin. A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing*, 9(1):3–12, Jan. 2005.

[9] I. Loshchilov, M. Schoenauer, and M. Sebag. Self-adaptive surrogate-assisted covariance matrix adaptation evolution strategy. In *Proceedings of the 14th GECCO*, GECCO '12, pages 321–328, New York, NY, USA, 2012. ACM.

[10] Y. S. Ong, P. B. Nair, and A. J. Keane. Evolutionary optimization of computationally expensive problems via surrogate modeling. *AIAA Journal*, 41(4):687–696, 2003.

[11] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning.* Adaptative computation and machine learning series. MIT Press, 2006.

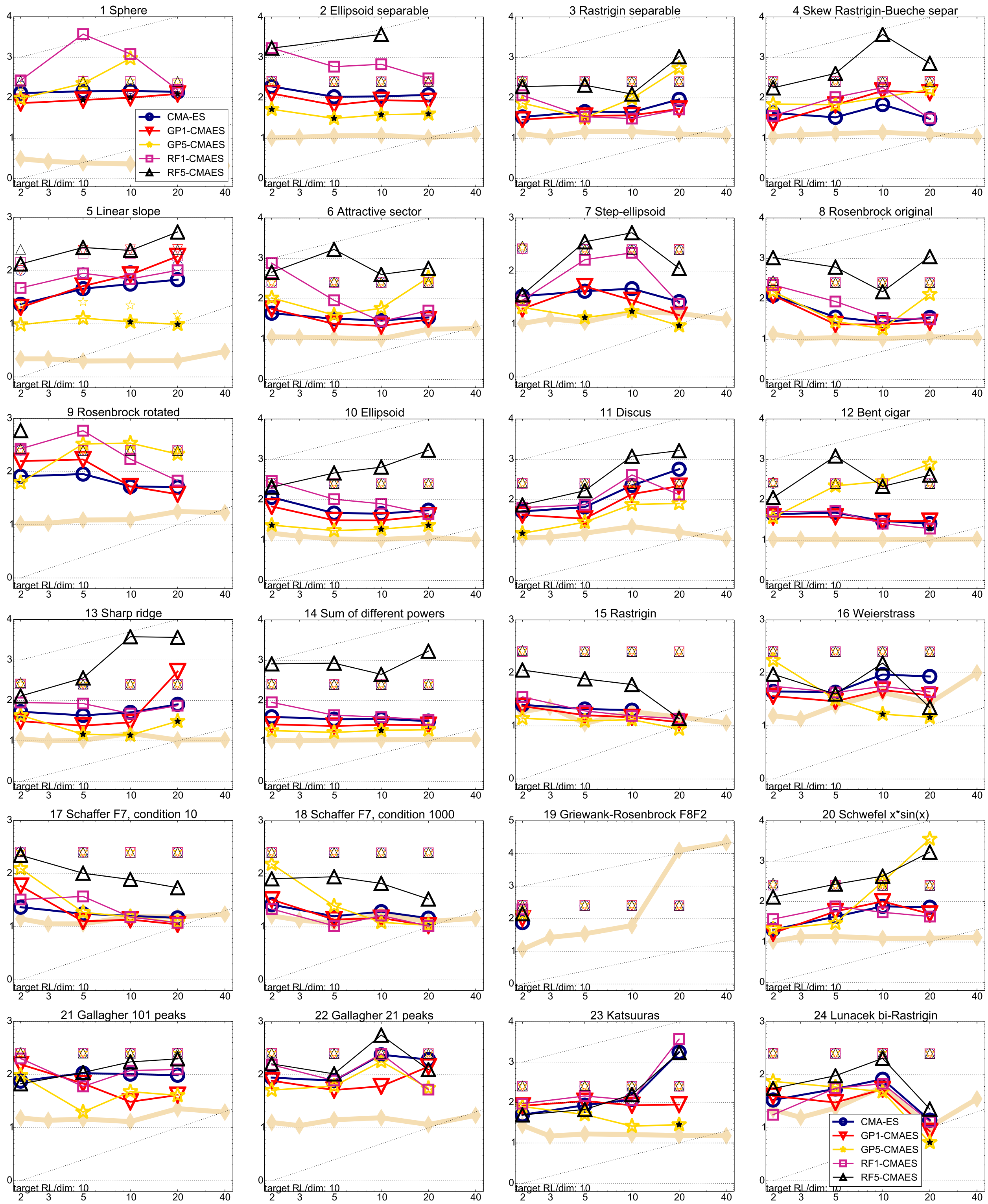

Figure 1: Expected running time (ERT in number of f-evaluations as $\log_{10}$ value) divided by dimension versus dimension. The target function value is chosen such that the bestGECCO2009 artificial algorithm just failed to achieve an ERT of $10 \times \mathrm{DIM}$. Different symbols correspond to different algorithms given in the legend of f_1 and f_{24}. Light symbols give the maximum number of function evaluations from the longest trial divided by dimension. Black stars indicate a statistically better result compared to all other algorithms with $p < 0.01$ and Bonferroni correction number of dimensions (six). Legend: ○:CMA-ES, ▽:GP1-CMAES, ⋆:GP5-CMAES, □:RF1-CMAES, △:RF5-CMAES

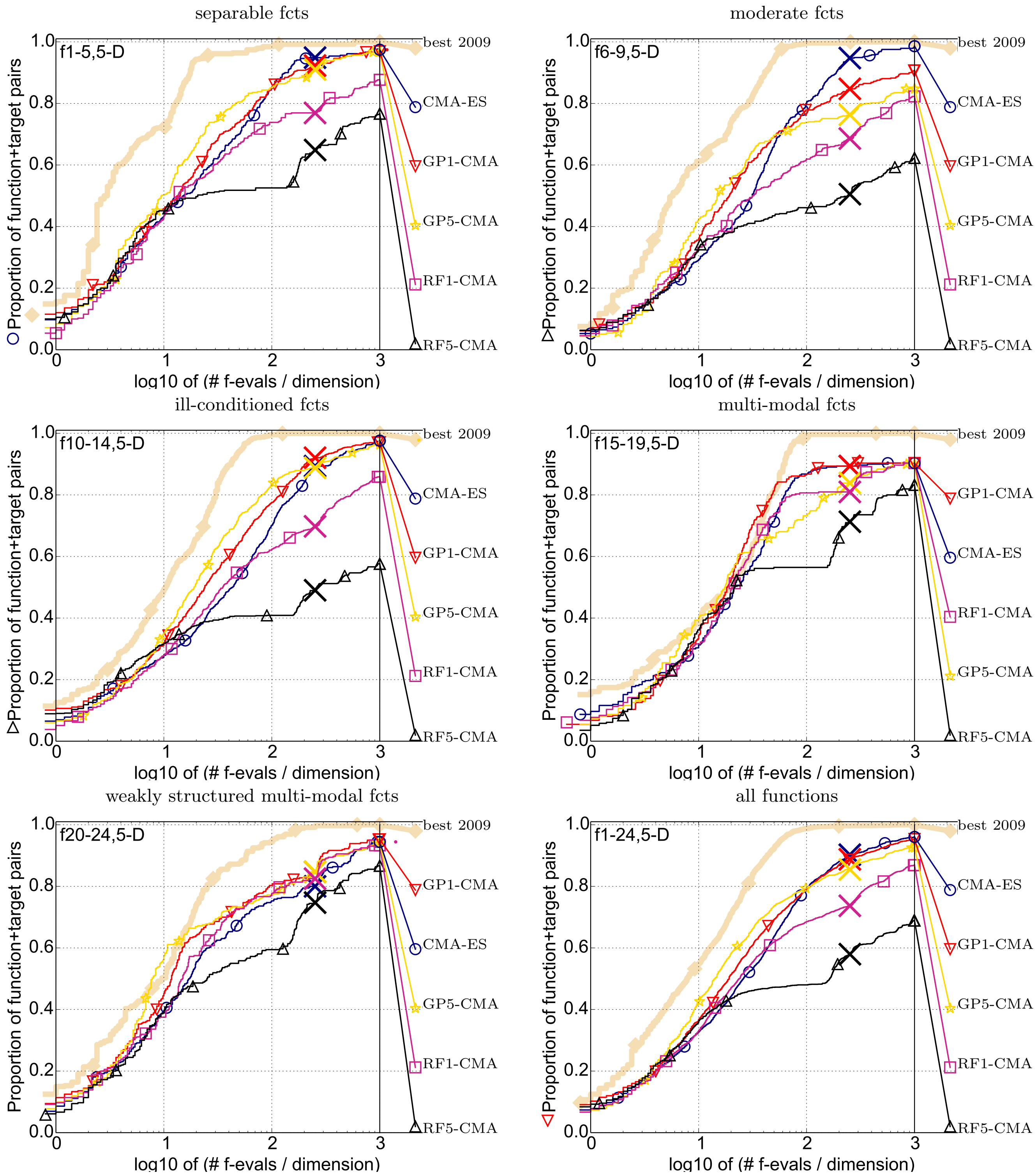

Figure 2: Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for all functions and subgroups in 5-D. The targets are chosen from $10^{[-8..2]}$ such that the bestGECCO2009 artificial algorithm just not reached them within a given budget of $k \times$ DIM, with $k \in \{0.5, 1.2, 3, 10, 50\}$. The "best 2009" line corresponds to the best ERT observed during BBOB 2009 for each selected target.

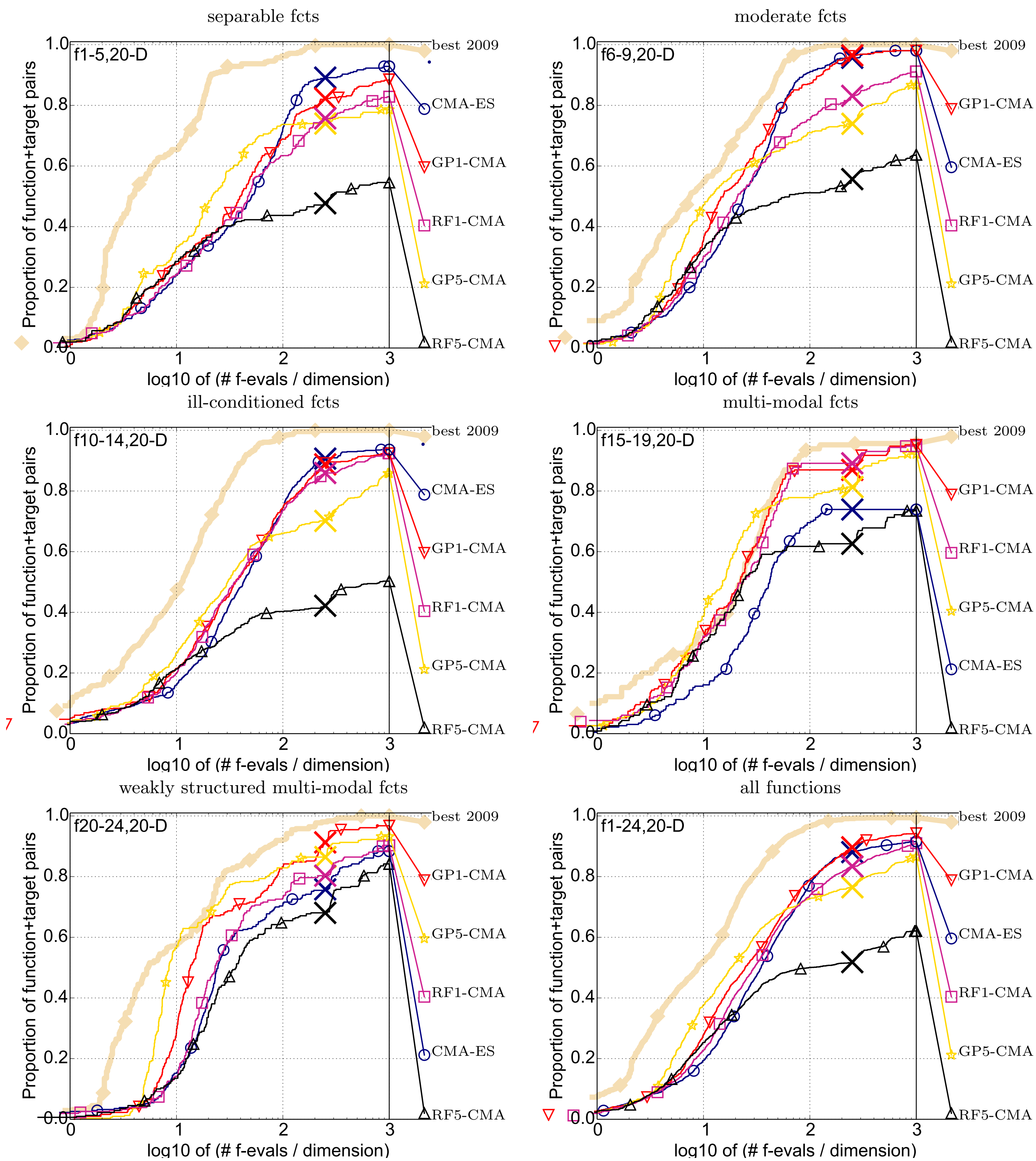

Figure 3: Bootstrapped empirical cumulative distribution of the number of objective function evaluations divided by dimension (FEvals/DIM) for all functions and subgroups in 20-D. The targets are chosen from $10^{[-8..2]}$ such that the bestGECCO2009 artificial algorithm just not reached them within a given budget of $k \times$ DIM, with $k \in \{0.5, 1.2, 3, 10, 50\}$. The "best 2009" line corresponds to the best ERT observed during BBOB 2009 for each selected target.

#FEs/D	0.5	1.2	3	10	50	#succ
f1	*2.5e+1*:4.8	*1.6e+1*:7.6	*1.0e-8*:12	*1.0e-8*:12	*1.0e-8*:12	15/15
CMA-ES	2.8(5)	2.9(1)	59(6)	59(6)	59(6)	45/45
GP1-CMA	2.5(2)	2.2(1)	**36**(8)$^{\star 2}$	**36**(6)$^{\star 2}$	**36**(3)$^{\star 2}$	15/15
GP5-CMA	2.3(2)	**2.1**(1.0)	92(67)	92(106)	92(35)	11/15
RF1-CMA	3.7(2)	3.0(2)	1520(1753)	1520(1366)	1520(1263)	1/15
RF5-CMA	**2.2**(3)	2.4(1)	∞	∞	∞ *1252*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f2	*1.6e+6*:2.9	*4.0e+5*:11	*4.0e+4*:15	*6.3e+2*:58	*1.0e-8*:95	15/15
CMA-ES	2.5(5)	**1.5**(1)	5.4(5)	9.2(4)	∞ *1258*	0/45
GP1-CMA	**2.1**(6)	1.9(2)	4.0(4)	5.8(4)	∞ *1258*	0/15
GP5-CMA	2.5(2)	1.7(2)	**3.2**(2)	**2.7**(0.8)*	**94**(70)$^{\star 3}$	2/15
RF1-CMA	2.9(4)	2.7(3)	6.8(4)	51(100)	∞ *1258*	0/15
RF5-CMA	2.1(2)	7.1(39)	25(28)	∞	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f3	*1.6e+2*:4.1	*1.0e+2*:15	*6.3e+1*:23	*2.5e+1*:73	*1.0e+1*:716	15/15
CMA-ES	3.2(5)	1.8(2)	2.6(1)	3.1(1)	**1.1**(0.5)	37/45
GP1-CMA	2.8(2)	1.5(2)	2.3(1)	2.5(3)	1.6(2)	11/15
GP5-CMA	3.0(3)	**1.3**(0.7)	**1.6**(1)	**2.2**(0.8)	2.6(2)	8/15
RF1-CMA	**1.9**(1)	1.6(1)	2.2(0.2)	2.4(1)	3.0(3)	6/15
RF5-CMA	3.4(3)	1.6(1)	4.6(10)	14(11)	6.1(8)	4/15

#FEs/D	0.5	1.2	3	10	50	#succ
f4	*2.5e+2*:2.6	*1.6e+2*:10	*1.0e+2*:19	*4.0e+1*:65	*1.6e+1*:434	15/15
CMA-ES	3.5(7)	2.4(1)	2.8(2)	**2.5**(0.7)	**1.1**(0.9)*	43/45
GP1-CMA	2.7(3)	2.4(3)	2.6(3)	5.1(7)	3.2(2)	9/15
GP5-CMA	**1.6**(1)	**1.6**(2)	3.0(3)	5.2(1)	7.5(5)	5/15
RF1-CMA	2.4(3)	1.9(2)	**2.5**(2)	7.9(15)	12(5)	3/15
RF5-CMA	2.7(4)	1.9(1)	4.9(2)	31(36)	∞ *1252*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f5	*6.3e+1*:4.0	*4.0e+1*:10	*1.0e-8*:10	*1.0e-8*:10	*1.0e-8*:10	15/15
CMA-ES	2.4(0.9)	1.9(1)	23(13)	23(14)	23(10)	45/45
GP1-CMA	**2.0**(2)	**1.5**(0.8)	26(59)	26(48)	26(36)	15/15
GP5-CMA	3.0(2)	1.7(0.1)	**6.4**(3)	**6.4**(2)	**6.4**(5)	15/15
RF1-CMA	2.3(1)	1.7(1)	45(43)	45(37)	45(46)	15/15
RF5-CMA	2.8(2)	1.9(1)	137(84)	137(87)	137(143)	10/15

#FEs/D	0.5	1.2	3	10	50	#succ
f6	*1.0e+5*:3.0	*2.5e+4*:8.4	*1.0e+2*:16	*2.5e+1*:54	*2.5e-1*:254	15/15
CMA-ES	3.9(4)	2.0(3)	3.2(3)	3.0(1)	**2.1**(0.4)$^{\star 6}$	45/45
GP1-CMA	**2.2**(3)	**1.4**(1)	**2.3**(2)	**2.2**(2)	74(62)	1/15
GP5-CMA	3.0(3)	1.8(1)	2.8(0.6)	3.7(1)	∞ *1260*	0/15
RF1-CMA	3.0(3)	2.0(2)	4.5(7)	8.5(12)	∞ *1258*	0/15
RF5-CMA	3.6(4)	2.4(2)	19(26)	154(164)	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f7	*1.6e+2*:4.2	*1.0e+2*:6.2	*2.5e+1*:20	*4.0e+0*:54	*1.0e+0*:324	15/15
CMA-ES	2.6(6)	3.1(1)	3.4(2)	3.8(1)	1.7(0.9)	43/45
GP1-CMA	**1.7**(0.9)	**2.0**(1)	2.1(1)	4.7(4)	1.4(1)	15/15
GP5-CMA	2.2(3)	2.5(2)	**1.9**(0.6)	**1.2**(0.4)*	**0.82**(0.7)	15/15
RF1-CMA	1.9(2)	2.6(1)	2.7(2)	15(25)	10(6)	5/15
RF5-CMA	2.5(2)	2.7(2)	5.3(13)	33(37)	17(26)	3/15

#FEs/D	0.5	1.2	3	10	50	#succ
f8	*1.0e+4*:4.6	*6.3e+3*:6.8	*1.0e+3*:18	*6.3e+1*:54	*1.6e+0*:258	15/15
CMA-ES	3.0(4)	2.8(2)	2.7(2)	3.2(3)	**4.1**(2)	37/45
GP1-CMA	2.8(3)	2.2(3)	2.3(2)	**2.1**(0.5)	7.2(10)	8/15
GP5-CMA	**2.1**(0.9)	**1.9**(2)	**1.9**(1)	2.5(2)	70(89)	1/15
RF1-CMA	2.6(3)	2.3(2)	3.1(2)	7.8(8)	36(49)	2/15
RF5-CMA	2.3(2)	2.0(2)	2.8(1)	56(104)	∞ *1252*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f9	*2.5e+1*:20	*1.6e+1*:26	*1.0e+1*:35	*4.0e+0*:62	*1.6e-2*:256	15/15
CMA-ES	**10**(2)	**8.0**(5)	**7.0**(0.9)	**7.3**(8)	**44**(33)	5/45
GP1-CMA	12(4)	10(3)	8.2(3)	14(23)	∞ *1258*	0/15
GP5-CMA	15(5)	14(2)	13(11)	27(35)	70(141)	1/15
RF1-CMA	38(51)	36(24)	30(21)	48(66)	∞ *1258*	0/15
RF5-CMA	106(199)	126(108)	257(218)	∞	∞ *1252*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f10	*2.5e+6*:2.9	*6.3e+5*:7.0	*2.5e+5*:17	*6.3e+3*:54	*2.5e+1*:297	15/15
CMA-ES	1.9(1)	2.0(1)	1.5(2)	4.4(3)	4.4(5)	32/45
GP1-CMA	**1.2**(0.5)	**0.94**(0.4)	**0.85**(1)	2.9(2)	2.0(1)	15/15
GP5-CMA	2.4(3)	1.6(1)	1.0(1)	**1.6**(0.7)	**0.95**(0.2)$^{\star 3}$	15/15
RF1-CMA	3.2(3)	2.3(2)	1.6(1)	10(13)	63(23)	1/15
RF5-CMA	2.7(2)	1.7(3)	4.5(0.3)	43(48)	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f11	*1.0e+6*:3.0	*6.3e+4*:6.2	*6.3e+2*:16	*6.3e+1*:74	*6.3e-1*:298	15/15
CMA-ES	2.0(1)	2.6(2)	4.7(3)	4.4(4)	63(58)	3/45
GP1-CMA	**1.4**(1)	2.4(2)	4.5(2)	2.3(0.8)	5.1(2)	11/15
GP5-CMA	1.5(2)	2.5(1)	**2.9**(1)	**1.9**(1.0)	**2.3**(0.3)$^{\star 2}$	14/15
RF1-CMA	2.2(2)	2.6(2)	3.9(3)	5.1(4)	∞ *1258*	0/15
RF5-CMA	2.2(3)	**2.3**(3)	3.0(2)	11(7)	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f12	*4.0e+7*:3.6	*1.6e+7*:7.6	*4.0e+6*:19	*1.6e+4*:52	*1.0e+0*:268	15/15
CMA-ES	2.2(4)	2.5(3)	2.8(2)	4.6(0.7)	9.3(5)	18/45
GP1-CMA	2.0(3)	2.5(2)	**1.9**(1)	**3.7**(2)	**6.2**(6)	8/15
GP5-CMA	3.1(4)	2.9(2)	3.0(3)	21(30)	8.5(9)	6/15
RF1-CMA	2.1(2)	3.2(4)	3.2(2)	5.1(2)	22(47)	3/15
RF5-CMA	**1.9**(2)	**1.7**(2)	5.6(20)	117(340)	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f13	*1.0e+3*:2.8	*6.3e+2*:8.4	*4.0e+2*:17	*6.3e+1*:52	*6.3e-2*:264	15/15
CMA-ES	2.9(4)	2.1(2)	2.9(3)	4.1(1)	13(10)	16/45
GP1-CMA	**2.3**(2)	**1.6**(1)	2.0(0.6)	2.3(0.7)	70(68)	1/15
GP5-CMA	3.2(3)	2.1(2)	**1.6**(0.4)	**1.4**(0.8)$^{\star 2}$	**11**(20)	5/15
RF1-CMA	3.2(3)	2.0(2)	2.4(2)	8.1(14)	69(111)	1/15
RF5-CMA	3.5(4)	2.2(1)	2.1(0.7)	35(65)	∞ *1252*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f14	*1.6e+1*:3.0	*1.0e+1*:10	*6.3e+0*:15	*2.5e-1*:53	*1.0e-5*:251	15/15
CMA-ES	3.5(3)	1.9(3)	2.1(2)	3.3(0.6)	**20**(11)$^{\star 2}$	11/45
GP1-CMA	3.2(5)	1.6(1)	1.9(1.0)	2.2(0.6)	∞ *1258*	0/15
GP5-CMA	4.0(3)	1.8(1)	**1.5**(1)	**1.6**(0.3)	∞ *1260*	0/15
RF1-CMA	3.2(3)	2.1(2)	2.8(2)	4.2(2)	∞ *1258*	0/15
RF5-CMA	**2.8**(2)	**1.2**(1)	2.0(0.6)	81(115)	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f15	*1.6e+2*:3.0	*1.0e+2*:13	*6.3e+1*:24	*4.0e+1*:55	*1.6e+1*:289	5/5
CMA-ES	**2.6**(3)	**1.4**(1)	1.8(1)	1.9(0.7)	1.5(2)	43/45
GP1-CMA	4.6(3)	1.9(2)	1.9(1)	1.4(0.8)	**1.2**(0.8)	15/15
GP5-CMA	3.5(4)	1.6(1)	**1.6**(1)	**1.1**(0.8)	3.0(3)	11/15
RF1-CMA	5.4(4)	2.2(2)	2.4(1)	1.7(1.0)	1.3(1)	14/15
RF5-CMA	3.5(5)	1.6(1)	2.4(1)	7.0(4)	5.7(10)	8/15

#FEs/D	0.5	1.2	3	10	50	#succ
f16	*4.0e+1*:4.8	*2.5e+1*:16	*1.6e+1*:46	*1.0e+1*:120	*4.0e+0*:334	15/15
CMA-ES	1.4(1)	1.3(1)	1.7(2)	1.8(1)	1.9(2)	40/45
GP1-CMA	1.6(1)	1.3(2)	**0.90**(0.8)	**1.2**(1)	**1.4**(2)	13/15
GP5-CMA	2.0(2)	1.5(2)	2.7(4)	1.3(1)	1.9(1)	13/15
RF1-CMA	**1.3**(1.0)	**0.90**(2)	1.3(0.7)	1.8(2)	2.1(3)	11/15
RF5-CMA	2.5(2)	1.3(1)	1.1(1)	1.7(0.5)	3.2(3)	9/15

#FEs/D	0.5	1.2	3	10	50	#succ
f17	*1.0e+1*:5.2	*6.3e+0*:26	*4.0e+0*:57	*2.5e+0*:110	*6.3e-1*:412	15/15
CMA-ES	3.2(5)	1.7(2)	1.5(0.7)	1.2(0.5)	0.72(0.5)	44/45
GP1-CMA	4.5(3)	**1.6**(1.0)	**1.1**(0.6)	**0.78**(0.3)	**0.45**(0.2)	15/15
GP5-CMA	3.6(2)	1.6(2)	1.6(5)	1.8(6)	2.5(2)	10/15
RF1-CMA	**3.0**(3)	2.2(2)	3.3(0.7)	3.9(3)	4.1(4)	7/15
RF5-CMA	4.8(5)	8.6(16)	9.0(9)	10(10)	22(48)	2/15

#FEs/D	0.5	1.2	3	10	50	#succ
f18	*6.3e+1*:3.4	*4.0e+1*:7.2	*2.5e+1*:20	*1.6e+1*:58	*1.6e+0*:318	15/15
CMA-ES	2.7(3)	2.6(3)	2.0(1)	1.4(0.9)	**1.1**(0.4)	44/45
GP1-CMA	2.4(3)	3.6(4)	2.1(1)	1.2(0.5)	1.2(0.6)	14/15
GP5-CMA	1.7(0.9)	3.0(4)	5.2(8)	2.1(3)	5.8(10)	7/15
RF1-CMA	**1.5**(2)	**1.7**(2)	**1.3**(0.7)	**0.89**(0.8)	2.2(2)	11/15
RF5-CMA	2.2(2)	3.0(3)	9.5(20)	7.6(18)	29(24)	2/15

#FEs/D	0.5	1.2	3	10	50	#succ
f19	*1.6e-1*:172	*1.0e-1*:242	*6.3e-2*:675	*4.0e-2*:3078	*2.5e-2*:4946	15/15
CMA-ES	∞	∞	∞	∞	∞ *1258*	0/45
GP1-CMA	∞	∞	∞	∞	∞ *1260*	0/15
GP5-CMA	∞	∞	∞	∞	∞ *1262*	0/15
RF1-CMA	∞	∞	∞	∞	∞ *1258*	0/15
RF5-CMA	∞	∞	∞	∞	∞ *1262*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f20	*6.3e+3*:5.1	*4.0e+3*:8.4	*4.0e+1*:15	*2.5e+0*:69	*1.0e+0*:851	15/15
CMA-ES	2.5(1)	2.1(2)	3.7(2)	3.1(0.9)	13(12)	5/45
GP1-CMA	**1.9**(2)	1.8(1)	3.0(1)	4.2(10)	**11**(10)	2/15
GP5-CMA	2.5(3)	1.8(2)	**2.1**(0.6)	**2.1**(1)	∞ *1260*	0/15
RF1-CMA	2.9(1)	2.4(3)	3.9(2)	5.6(6)	∞ *1258*	0/15
RF5-CMA	2.5(2)	**1.8**(1)	25(30)	20(28)	∞ *1260*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f21	*4.0e+1*:3.9	*2.5e+1*:11	*1.6e+1*:31	*6.3e+0*:73	*1.6e+0*:347	5/5
CMA-ES	2.2(5)	1.7(2)	1.5(0.9)	7.3(10)	10(16)	13/45
GP1-CMA	**1.1**(1)	**1.4**(1.0)	0.99(0.6)	4.7(13)	4.9(13)	7/15
GP5-CMA	2.5(2)	1.7(1)	**0.89**(0.4)	**1.4**(1)	**4.7**(4)	8/15
RF1-CMA	2.0(2)	1.8(3)	1.4(1)	4.1(14)	4.8(7)	7/15
RF5-CMA	1.8(1)	2.6(2)	2.9(7)	7.5(9)	10(7)	5/15

#FEs/D	0.5	1.2	3	10	50	#succ
f22	*6.3e+1*:3.6	*4.0e+1*:15	*2.5e+1*:32	*1.0e+1*:71	*1.6e+0*:341	5/5
CMA-ES	3.4(5)	2.3(2)	1.7(1.0)	5.5(14)	8.9(9)	14/45
GP1-CMA	2.6(3)	**1.4**(1)	**1.2**(0.8)	**3.6**(9)	10(9)	4/15
GP5-CMA	2.9(2)	7.5(24)	4.6(12)	4.3(10)	11(13)	4/15
RF1-CMA	3.1(3)	2.0(2)	1.3(1)	5.5(27)	**3.2**(8)	9/15
RF5-CMA	**2.1**(4)	4.7(2)	2.9(0.1)	7.3(8)	24(14)	2/15

#FEs/D	0.5	1.2	3	10	50	#succ
f23	*1.0e+1*:3.0	*6.3e+0*:9.0	*4.0e+0*:33	*2.5e+0*:84	*1.0e+0*:518	15/15
CMA-ES	2.5(2)	2.0(3)	3.1(6)	5.2(4)	26(31)	4/45
GP1-CMA	1.9(2)	2.7(5)	3.2(3)	6.5(3)	4.9(5)	6/15
GP5-CMA	2.4(2)	2.0(1)	**2.4**(2)	**3.0**(4)	**2.2**(2)	11/15
RF1-CMA	**1.8**(2)	2.0(0.9)	3.0(3)	8.7(12)	∞ *1260*	0/15
RF5-CMA	2.4(2)	**1.5**(1)	3.3(3)	4.0(4)	∞ *1288*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f24	*6.3e+1*:15	*4.0e+1*:37	*4.0e+1*:37	*2.5e+1*:118	*1.6e+1*:692	15/15
CMA-ES	2.3(3)	2.4(2)	2.4(2)	2.3(1)	1.6(4)	33/45
GP1-CMA	1.7(1)	1.7(1)	1.7(1)	**1.3**(0.4)	**1.1**(1.0)	13/15
GP5-CMA	**1.5**(0.8)	**1.1**(0.5)	**1.1**(0.4)	2.5(3)	1.6(2)	9/15
RF1-CMA	1.9(2)	2.4(1)	2.4(3)	2.4(4)	1.9(1)	9/15
RF5-CMA	1.8(2)	3.6(11)	3.6(2)	4.1(2)	2.9(5)	7/15

Table 2: Expected running time (ERT in number of function evaluations) divided by the respective best ERT measured during BBOB-2009 in dimension 5. The ERT and in braces, as dispersion measure, the half difference between 90 and 10%-tile of bootstrapped run lengths appear for each algorithm and run-length based target, the corresponding best ERT (preceded by the target Δf-value in *italics*) in the first row. #succ is the number of trials that reached the target value of the last column. The median number of conducted function evaluations is additionally given in *italics*, if the target in the last column was never reached. Entries, succeeded by a star, are statistically significantly better (according to the rank-sum test) when compared to all other algorithms of the table, with $p = 0.05$ or $p = 10^{-k}$ when the number k following the star is larger than 1, with Bonferroni correction by the number of instances.

#FEs/D	0.5	1.2	3	10	50	#succ
f1	*6.3e+1*:24	*4.0e+1*:42	*1.0e-8*:43	*1.0e-8*:43	*1.0e-8*:43	15/15
CMA-ES	4.9(2)	4.5(0.8)	64(2)	64(3)	64(3)	45/45
GP1-CMA	3.9(1)	3.1(0.5)	**58**(9)$^{\star 2}$	**58**(5)$^{\star 2}$	**58**(7)$^{\star 2}$	15/15
GP5-CMA	**2.9**(0.3)	**2.0**(0.2)*	∞	∞	∞ *5034*	0/15
RF1-CMA	3.9(2)	3.5(0.7)	73(17)	73(17)	73(11)	15/15
RF5-CMA	3.7(1)	3.0(2)	∞	∞	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f2	*4.0e+6*:29	*2.5e+6*:42	*1.0e+5*:65	*1.0e+4*:207	*1.0e-8*:412	15/15
CMA-ES	1.0(0.7)	1.3(0.4)	14(4)	11(2)	∞ *5006*	0/45
GP1-CMA	1.5(0.4)	1.6(1)	8.9(6)	8.0(3)	∞ *5006*	0/15
GP5-CMA	**0.90**(1.0)	1.3(1)	**5.4**(1)$^{\star 2}$	**3.9**(0.5)$^{\star 3}$	∞ *5006*	0/15
RF1-CMA	1.1(0.8)	**1.2**(1)	11(5)	29(12)	∞ *5006*	0/15
RF5-CMA	1.2(1)	1.3(1.0)	184(109)	∞	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f3	*6.3e+2*:33	*4.0e+2*:44	*1.6e+2*:109	*1.0e+2*:255	*2.5e+1*:3277	15/15
CMA-ES	2.0(1)	4.0(1)	7.5(2)	7.2(2)	**7.8**(8)	8/45
GP1-CMA	2.3(0.6)	3.2(1)	**5.9**(2)	4.2(4)	22(19)	1/15
GP5-CMA	1.9(1.0)	2.9(1)	15(10)	43(65)	∞ *5034*	0/15
RF1-CMA	2.0(2)	3.7(2)	6.4(2)	**4.0**(0.7)	22(15)	1/15
RF5-CMA	**1.7**(0.9)	**2.6**(0.7)	18(25)	82(75)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f4	*6.3e+2*:22	*4.0e+2*:91	*2.5e+2*:250	*1.6e+2*:332	*6.3e+1*:1927	15/15
CMA-ES	7.7(2)	3.4(1)	**2.4**(0.6)	**4.1**(0.8)	**2.4**(4)$^{\star 6}$	30/45
GP1-CMA	9.2(7)	4.5(0.8)	11(13)	42(68)	39(45)	1/15
GP5-CMA	**7.5**(2)	5.5(1)	13(10)	215(178)	∞ *5022*	0/15
RF1-CMA	8.1(2)	**3.4**(1)	2.5(0.9)	7.7(11)	∞ *5006*	0/15
RF5-CMA	8.9(3)	13(15)	57(75)	∞	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f5	*2.5e+2*:19	*1.6e+2*:34	*1.0e-8*:41	*1.0e-8*:41	*1.0e-8*:41	15/15
CMA-ES	1.6(1)	1.9(0.8)	33(9)	33(15)	33(15)	45/45
GP1-CMA	**1.6**(1)	1.9(0.5)	92(83)	92(74)	92(83)	11/15
GP5-CMA	1.7(0.6)	**1.7**(0.4)	**4.8**(2)$^{\star 4}$	**4.8**(1)$^{\star 4}$	**4.8**(0.6)$^{\star 4}$	15/15
RF1-CMA	1.8(0.6)	2.3(1)	50(21)	50(31)	50(34)	15/15
RF5-CMA	2.0(1)	2.0(0.7)	265(494)	265(415)	265(166)	6/15

#FEs/D	0.5	1.2	3	10	50	#succ
f6	*2.5e+5*:16	*6.3e+4*:43	*1.6e+4*:62	*1.6e+2*:353	*1.6e+1*:1078	15/15
CMA-ES	3.0(2)	2.4(2)	2.7(2)	2.0(0.4)	**1.5**(0.2)$^{\star 4}$	45/45
GP1-CMA	2.8(2)	2.0(1)	2.0(0.4)	**1.7**(0.5)	4.2(4)	11/15
GP5-CMA	2.6(1)	1.6(0.5)	**1.5**(0.3)	19(20)	∞ *5024*	0/15
RF1-CMA	2.7(2)	2.1(0.9)	2.3(0.4)	2.9(1)	66(36)	1/15
RF5-CMA	**2.1**(2)	**1.6**(1)	1.8(0.6)	32(41)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f7	*1.0e+3*:11	*4.0e+2*:39	*2.5e+2*:74	*6.3e+1*:319	*1.0e+1*:1351	15/15
CMA-ES	**1.7**(1)	2.4(2)	2.6(1)	1.7(0.5)	**1.1**(0.3)	45/45
GP1-CMA	2.1(2)	2.2(1)	1.7(0.5)	0.92(0.3)	3.0(3)	10/15
GP5-CMA	2.4(1)	**1.8**(0.4)	**1.3**(0.3)	**0.58**(0.1)$^{\star 2}$	1.6(2)	14/15
RF1-CMA	2.0(2)	2.6(2)	2.3(1)	1.5(0.4)	54(87)	1/15
RF5-CMA	1.8(2)	1.9(0.5)	1.8(0.4)	7.0(5)	∞ *5034*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f8	*4.0e+4*:19	*2.5e+4*:35	*4.0e+3*:67	*2.5e+2*:231	*1.6e+1*:1470	15/15
CMA-ES	6.5(3)	4.7(0.9)	4.5(0.7)	3.0(1.0)	3.6(1)	31/45
GP1-CMA	5.5(2)	3.4(0.7)	3.1(0.4)	**2.3**(0.4)	**3.0**(1)	12/15
GP5-CMA	**4.4**(0.7)	**2.6**(0.2)*	**2.2**(0.7)$^{\star 2}$	11(8)	8.2(4)	5/15
RF1-CMA	5.8(2)	4.0(0.8)	4.0(0.4)	2.7(1.0)	25(34)	2/15
RF5-CMA	5.8(1)	3.7(1)	4.2(2)	96(109)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f9	*1.0e+2*:357	*6.3e+1*:560	*4.0e+1*:684	*2.5e+1*:756	*1.0e+1*:1716	15/15
CMA-ES	2.9(0.3)	2.5(0.2)	2.2(3)	2.2(0.5)	∞ *5006*	0/45
GP1-CMA	**2.1**(0.8)	**1.6**(0.3)	**1.4**(0.6)	**1.5**(0.1)	∞ *5006*	0/15
GP5-CMA	12(28)	12(25)	13(19)	12(4)	∞ *5020*	0/15
RF1-CMA	3.8(1)	5.8(4)	5.3(10)	5.8(6)	∞ *5006*	0/15
RF5-CMA	∞	∞	∞	∞	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f10	*1.6e+6*:15	*1.0e+6*:27	*4.0e+5*:70	*6.3e+4*:231	*4.0e+3*:1015	15/15
CMA-ES	5.6(6)	5.5(4)	5.1(2)	4.8(1)	3.2(0.9)	45/45
GP1-CMA	5.4(4)	5.4(2)	4.2(0.9)	3.5(2)	2.3(0.8)	15/15
GP5-CMA	**4.8**(2)	**3.6**(1)	**3.1**(2)	**2.0**(0.3)$^{\star 2}$	**1.1**(0.4)$^{\star 3}$	15/15
RF1-CMA	5.9(5)	5.3(2)	4.1(0.9)	3.9(2)	74(97)	1/15
RF5-CMA	5.1(3)	5.2(2)	10(19)	145(217)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f11	*4.0e+4*:11	*2.5e+3*:27	*1.6e+2*:313	*1.0e+2*:481	*1.0e+1*:1002	15/15
CMA-ES	1.7(1)	2.3(2)	36(35)	151(213)	∞ *5006*	0/45
GP1-CMA	**1.3**(1)	**1.8**(3)	14(5)	15(19)	∞ *5006*	0/15
GP5-CMA	1.8(1)	2.3(2)	**5.2**(3)	17(35)	∞ *5008*	0/15
RF1-CMA	1.5(3)	3.2(3)	8.6(5)	**12**(5)	∞ *5006*	0/15
RF5-CMA	1.9(2)	2.8(2)	105(100)	146(133)	∞ *5008*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f12	*1.0e+8*:23	*6.3e+7*:39	*2.5e+7*:76	*4.0e+6*:209	*1.0e+1*:1042	15/15
CMA-ES	5.5(1)	4.9(1)	3.8(0.9)	2.4(0.4)	3.4(4)	35/45
GP1-CMA	4.7(3)	**3.9**(2)	3.7(0.9)	2.8(1)	**2.4**(0.2)	13/15
GP5-CMA	18(46)	22(18)	38(31)	74(96)	21(30)	3/15
RF1-CMA	**3.9**(2)	3.9(2)	**3.1**(0.8)	**1.8**(0.2)$^{\star 2}$	3.0(1)	12/15
RF5-CMA	5.2(3)	4.6(3)	9.4(19)	39(108)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f13	*1.6e+3*:28	*1.0e+3*:64	*6.3e+2*:79	*4.0e+1*:211	*2.5e+0*:1724	15/15
CMA-ES	3.5(0.9)	4.0(1.0)	5.3(1)	7.6(6)	6.4(3)	16/45
GP1-CMA	2.5(0.9)	2.4(0.4)	2.9(0.5)	49(50)	42(32)	1/15
GP5-CMA	**2.4**(0.9)	**1.5**(0.3)$^{\star 4}$	**1.6**(0.2)$^{\star 4}$	**2.9**(0.8)$^{\star 2}$	**4.5**(6)	7/15
RF1-CMA	3.2(1)	3.0(0.6)	3.9(1)	7.1(3)	7.3(7)	5/15
RF5-CMA	3.4(1)	3.0(0.8)	4.2(1)	343(297)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f14	*2.5e+1*:15	*1.6e+1*:42	*1.0e+1*:75	*1.6e+0*:219	*6.3e-4*:1106	15/15
CMA-ES	9.1(3)	4.9(2)	4.0(1)	2.9(0.4)	**4.0**(0.3)	44/45
GP1-CMA	7.9(5)	3.9(1)	3.0(0.7)	2.3(0.9)	4.6(4)	13/15
GP5-CMA	**5.7**(2)	**2.6**(0.6)	**2.1**(0.6)	**1.7**(0.4)	67(68)	1/15
RF1-CMA	7.1(3)	4.4(2)	3.5(1)	3.1(1)	33(32)	2/15
RF5-CMA	6.5(3)	3.9(2)	3.7(1)	153(144)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f15	*6.3e+2*:15	*4.0e+2*:67	*2.5e+2*:292	*1.6e+2*:846	*1.0e+2*:1671	15/15
CMA-ES	**3.6**(3)	1.9(0.5)	0.81(0.1)	**0.78**(0.2)	**0.64**(0.2)	15/15
GP1-CMA	3.7(2)	**1.5**(0.4)	**0.60**(0.1)	2.2(3)	3.9(7)	7/15
GP5-CMA	4.5(2)	2.2(0.6)	0.93(0.3)	0.92(0.4)	0.70(0.2)	15/15
RF1-CMA	3.6(0.9)	1.9(0.8)	0.95(0.5)	1.7(2)	6.5(6)	5/15

#FEs/D	0.5	1.2	3	10	50	#succ
f16	*4.0e+1*:26	*2.5e+1*:127	*1.6e+1*:540	*1.6e+1*:540	*1.0e+1*:1384	15/15
CMA-ES	3.3(3)	10(5)	3.2(1)	3.2(2)	1.4(0.6)	45/45
GP1-CMA	3.3(3)	4.3(1)	1.4(0.3)	1.4(0.4)	0.90(0.2)	14/15
GP5-CMA	3.4(2)	**1.6**(0.3)	**0.54**(0.1)$^{\star 2}$	**0.54**(0.1)$^{\star 2}$	**0.57**(0.7)	15/15
RF1-CMA	**2.8**(4)	4.6(3)	1.6(0.4)	1.6(0.3)	0.79(0.2)	15/15
RF5-CMA	4.2(5)	2.2(0.9)	0.83(0.2)	0.83(0.3)	1.0(2)	13/15

#FEs/D	0.5	1.2	3	10	50	#succ
f17	*1.6e+1*:11	*1.0e+1*:63	*6.3e+0*:305	*4.0e+0*:468	*1.0e+0*:1030	15/15
CMA-ES	3.3(2)	2.2(1)	0.96(0.4)	1.0(0.3)	**1.0**(0.3)	45/45
GP1-CMA	**2.4**(2)	**1.4**(0.7)	**0.73**(0.3)	**0.79**(0.3)	3.4(5)	10/15
GP5-CMA	3.3(3)	1.6(1.0)	0.79(0.3)	0.87(0.7)	11(20)	5/15
RF1-CMA	3.1(3)	1.9(0.8)	0.78(0.2)	0.83(0.5)	4.2(9)	9/15
RF5-CMA	4.0(4)	2.7(2)	3.6(4)	17(16)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f18	*4.0e+1*:116	*2.5e+1*:252	*1.6e+1*:430	*1.0e+1*:621	*4.0e+0*:1090	15/15
CMA-ES	1.1(0.5)	1.2(0.4)	1.1(0.9)	1.1(0.2)	**1.2**(0.2)	44/45
GP1-CMA	**0.81**(0.3)	0.84(0.1)	**0.85**(0.1)	**0.93**(0.4)	5.2(9)	8/15
GP5-CMA	0.95(0.2)	0.85(0.6)	1.7(3)	2.8(4)	19(21)	3/15
RF1-CMA	0.87(0.8)	**0.82**(0.3)	0.87(0.3)	1.0(0.5)	10(13)	5/15
RF5-CMA	1.3(1)	2.7(1)	5.4(3)	53(99)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f19	*1.6e-1*:2.5e5	*1.0e-1*:3.4e5	*6.3e-2*:3.4e5	*4.0e-2*:3.4e5	*2.5e-2*:3.4e5	3/15
CMA-ES	∞	∞	∞	∞	∞ *5008*	0/45
GP1-CMA	∞	∞	∞	∞	∞ *5006*	0/15
GP5-CMA	∞	∞	∞	∞	∞ *5020*	0/15
RF1-CMA	∞	∞	∞	∞	∞ *5008*	0/15
RF5-CMA	∞	∞	∞	∞	∞ *5034*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f20	*1.6e+4*:38	*1.0e+4*:42	*2.5e+2*:62	*2.5e+0*:250	*1.6e+0*:2536	15/15
CMA-ES	3.3(2)	4.0(1)	5.9(1)	5.8(1)	11(8)	7/45
GP1-CMA	2.8(1)	3.2(1)	3.6(0.7)	4.0(1)	∞ *5006*	0/15
GP5-CMA	**2.3**(0.3)	**2.2**(0.1)*	**2.5**(0.4)$^{\star 3}$	284(332)	∞ *5022*	0/15
RF1-CMA	3.5(1)	3.9(1)	5.2(2)	**3.4**(0.5)	**6.1**(4)	4/15
RF5-CMA	2.6(1)	2.9(1)	6.6(4)	134(160)	∞ *5006*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f21	*6.3e+1*:36	*4.0e+1*:77	*4.0e+1*:77	*1.6e+1*:456	*4.0e+0*:1094	15/15
CMA-ES	5.9(3)	7.5(6)	7.5(18)	4.3(6)	8.8(8)	17/45
GP1-CMA	3.8(2)	6.3(30)	6.3(1)	1.8(0.2)	**3.4**(3)	10/15
GP5-CMA	**3.2**(0.4)	**2.9**(0.8)	**2.9**(0.4)	**1.8**(2)	7.7(9)	6/15
RF1-CMA	5.2(2)	3.7(1)	3.7(1)	5.5(6)	8.0(13)	6/15
RF5-CMA	7.6(5)	6.1(3)	6.1(5)	8.7(9)	14(18)	4/15

#FEs/D	0.5	1.2	3	10	50	#succ
f22	*6.3e+1*:45	*4.0e+1*:68	*4.0e+1*:68	*1.6e+1*:231	*6.3e+0*:1219	15/15
CMA-ES	4.8(2)	10(17)	10(9)	17(22)	9.2(12)	15/45
GP1-CMA	12(30)	17(20)	17(15)	12(17)	2.7(5)	10/15
GP5-CMA	**2.5**(1)*	**3.6**(0.3)$^{\star 2}$	**3.6**(12)$^{\star 2}$	4.8(11)	**2.0**(2)	11/15
RF1-CMA	4.6(2)	5.2(3)	5.2(3)	**4.6**(2)	3.0(4)	10/15
RF5-CMA	5.8(2)	11(2)	11(21)	11(12)	5.7(15)	7/15

#FEs/D	0.5	1.2	3	10	50	#succ
f23	*6.3e+0*:29	*4.0e+0*:118	*2.5e+0*:306	*2.5e+0*:306	*1.0e+0*:1614	15/15
CMA-ES	1.4(0.6)	6.9(12)	115(67)	115(75)	∞ *5020*	0/45
GP1-CMA	2.1(2)	6.4(5)	5.8(4)	5.8(2)	1.6(1)	14/15
GP5-CMA	2.2(2)	**2.9**(0.6)	**1.9**(0.1)$^{\star 2}$	**1.9**(0.3)$^{\star 2}$	**0.84**(0.8)	15/15
RF1-CMA	**1.3**(1)	3.4(5)	244(119)	244(390)	∞ *5010*	0/15
RF5-CMA	1.8(2)	4.5(5)	113(109)	113(144)	∞ *5086*	0/15

#FEs/D	0.5	1.2	3	10	50	#succ
f24	*2.5e+2*:208	*1.6e+2*:918	*1.0e+2*:6628	*6.3e+1*:9885	*4.0e+1*:31629	15/15
CMA-ES	1.3(0.3)	1.7(1)	3.6(3)	3.2(2)	3.6(7)	2/45
GP1-CMA	0.83(0.1)	**0.84**(0.5)	**0.20**(0.0)	**0.24**(0.3)	**0.38**(0.4)	5/15
GP5-CMA	**0.51**(0.1)$^{\star 2}_{\downarrow 4}$	0.86(1)	0.37(0.6)	0.32(0.5)	1.1(0.8)	2/15
RF1-CMA	1.3(0.3)	2.0(1)	0.50(0.0)	0.52(0.6)	∞ *5006*	0/7
RF5-CMA	2.2(1.0)	4.1(4)	11(10)	∞	∞ *5034*	0/15

Table 3: Expected running time (ERT in number of function evaluations) divided by the respective best ERT measured during BBOB-2009 in dimension 20. The ERT and in braces, as dispersion measure, the half difference between 90 and 10%-tile of bootstrapped run lengths appear for each algorithm and run-length based target, the corresponding best ERT (preceded by the target Δf-value in *italics*) in the first row. #succ is the number of trials that reached the target value of the last column. The median number of conducted function evaluations is additionally given in *italics*, if the target in the last column was never reached. Entries, succeeded by a star, are statistically significantly better (according to the rank-sum test) when compared to all other algorithms of the table, with $p = 0.05$ or $p = 10^{-k}$ when the number k following the star is larger than 1, with Bonferroni correction by the number of instances.