Expensive Optimization Scenario: IPOP-CMA-ES with a Population Bound Mechanism for Noiseless Function Testbed

Tianjun Liao
IRIDIA, CoDE, Université Libre de Bruxelles (ULB), Brussels, Belgium
tliao@ulb.ac.be

Thomas Stützle
IRIDIA, CoDE, Université Libre de Bruxelles (ULB), Brussels, Belgium
stuetzle@ulb.ac.be

ABSTRACT

In this paper, we test the results from the default and tuned IPOP-CMA-ES with population bound mechanism (labeled as `def` and `texp`, respectively) [8, 9] on the expensive optimization scenario of the BBOB benchmark. In `texp` [9], seven parameters that directly control the internal parameters were tuned by applying an automatic algorithm configuration tool on the solution quality after $100 \times D$ function evaluations. We compare the results of `texp` to those of the default variant (`def`) [8,9] in the expensive optimization scenario. We find that `texp` often converges faster than `def`.

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

General Terms

Algorithms

Keywords

Benchmarking, Black-box optimization

1. INTRODUCTION

BBOB 2013 sets an expensive optimization scenario for the algorithms that are designed for the problems where only few function evaluations are affordable. In this expensive optimization scenario, a focus on the first $100 \times D$ function evaluations is assumed.

In this paper, we re-use the results from `def` and `texp`, taken from a companion submission to BBOB 2013 [8,9] and compare them under the expensive optimization scenario. To make the presentation here reasonably self-contained, we repeat here a concise explanation of the two algorithms, more details can also be found in [8, 9].

`def` is a variant of IPOP-CMA-ES where the only modification is that it uses a bound on the maximum population size that IPOP-CMA-ES may use. This bound is set to a value of $10 \cdot D^2$, where D is the dimension of the problem to be solved. Once this bound is reached, the population size is reset to its initial value, which in IPOP-CMA-ES is as default set to $\lambda = 4 + \lfloor 3\ln(D) \rfloor$ and the schedule of the population size increment in IPOP-CMA-ES is restarted. As shown in [8], the usage of this bound helps to improve IPOP-CMA-ES's performance on few of the hardest benchmark functions but it has (obviously) no effect on problems that are solved easily to optimality with IPOP-CMA-ES before reaching the population size bound.

`texp` is based on `def` but it uses an additional tuning of IPOP-CMA-ES parameters. In particular, we have used for the tuning the irace tool [10], a publicly available implementation of the automatic configuration method Iterated F-Race [1]. To keep a separation between training functions that are seen during the tuning and the test functions of the BBOB benchmark set, we have used for the tuning the functions that were defined for a special issue of the Soft Computing (SOCO) journal [5]. For the tuning we follow [6] and use as the tuning set these SOCO functions with dimensions taken as $D \in [5, 40]$. Even if more parameters of IPOP-CMA-ES may undergo further fine-tuning, here we limit the parameters to be tuned to those that we also considered in earlier work on the tuning of IPOP-CMA-ES [7]. The so obtained parameter settings are reproduced in Table 1; for more details we refer to [7]. Note that the tuning was done to reach best possible solution quality with $100 \cdot D$ function evaluations and not to minimize the time-to-target, which is the evaluation criterion actually measures in the BBOB benchmark set. Nevertheless, the tuning resulted in improvements over the default version, as it is also clear from the computational results that are presented in this paper.

2. RESULTS

Results of `def` and `texp` from experiments according to [3] on the benchmark functions given in [2, 4] are presented in Figures 2, 4 and 6 and in Tables 2 to 5.

We also show results of `def` in Figure 3 and Table 3 to compare with `texp`. Additional results are given in Figure 1, where for each of the two IPOP-CMA-ES variants we give a graphical representation of the data contained in Tables 2 and 3 for each of the five bounds and across all bounds. In each of the plots a point has as the x-coordinate the expected running time (ERT) divided by the best ERT mea-

GECCO'13 Companion, July 6–10, 2013, Amsterdam, The Netherlands.

Table 1: Parameters that have been considered for tuning. Given are the continuous range we considered for tuning. The last column gives the parameter settings obtained when tuning for the final solution quality at $100 \times D$ function evaluations (`texp`).

Parameter	Internal parameter	Range	`texp`
a	Init pop size: $\lambda_0 = 4 + \lfloor a \ln(D) \rfloor$	$[1, 10]$	2.675
b	Parent size: $\mu = \lfloor \lambda / b \rfloor$	$[1, 5]$	1.351
c	Init step size: $\sigma_0 = c \cdot (B - A)$	$(0, 1)$	0.102
d	IPOP factor: $ipop = d$	$[1, 4]$	2.88
e	$stopTolFun = 10^e$	$[-20, -6]$	-8.607
f	$stopTolFunHist = 10^f$	$[-20, -6]$	-14.77
g	$stopTolX = 10^g$	$[-20, -6]$	-9.529

sured during BBOB-2009 for `texp` and as y-coordinate the same measure for `def`. A point above the diagonal indicates that `texp` reaches the corresponding bound faster than `def`; a value below 1 indicates that an algorithm has a lower ERT than best algorithm from BBOB-2009 for a specific bound. These plots indicate that `texp` is faster in finding specific targets than `def` especially if the target values are not too strict.

3. ACKNOWLEDGMENTS

The authors would like to thank the great and hard work of the BBOB team. This work was supported by the Meta-X project funded by the Scientific Research Directorate of the French Community of Belgium. Thomas Stützle acknowledges support from the Belgian F.R.S.-FNRS, of which he is a Research Associate. Tianjun Liao acknowledges a fellowship from the China Scholarship Council.

4. REFERENCES

[1] P. Balaprakash, M. Birattari, and T. Stützle. Improvement strategies for the F-Race algorithm: sampling design and iterative refinement. In Bartz-Beielstein et al., editors, *Proceedings of the 4th International Conference on Hybrid Metaheuristics*, volume 4771 of *LNCS*, pages 108–122. Springer, Berlin, Germany, 2007.

[2] S. Finck, N. Hansen, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Presentation of the noiseless functions. Technical Report 2009/20, Research Center PPE, 2009. Updated February 2010.

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

[4] 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.

[5] F. Herrera, M. Lozano, and D. Molina. Test suite for the special issue of soft computing on scalability of evolutionary algorithms and other metaheuristics for large scale continuous optimization problems, 2010. URL: `http://sci2s.ugr.es/eamhco/`.

[6] T. Liao, M. A. Montes de Oca, and T. Stützle. Tuning parameters across mixed dimensional instances: A performance scalability study of Sep-G-CMA-ES. In

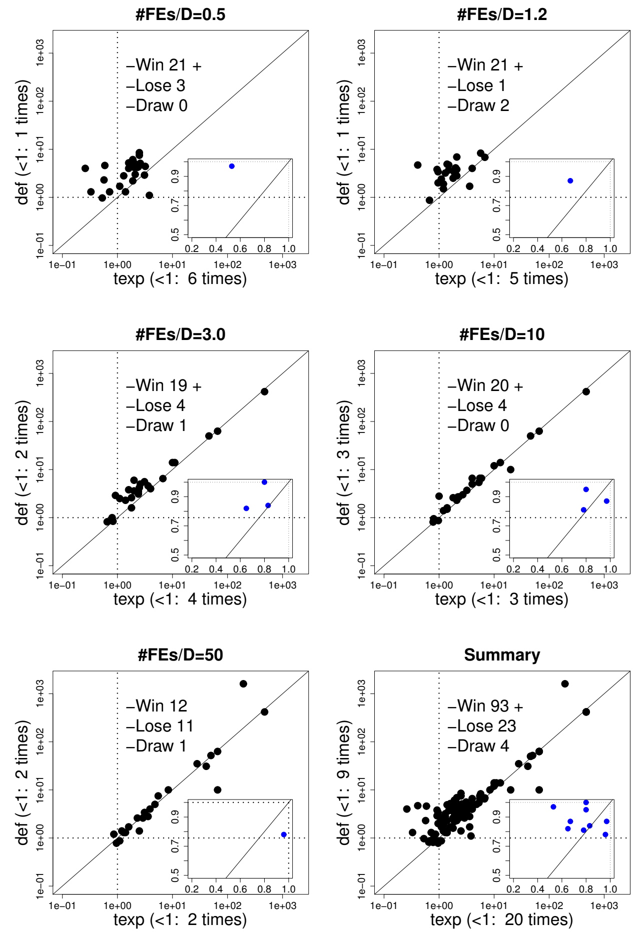


Figure 1: Correlation plot of `texp` and `def` in dimension 20. Given is for each bound on the target values (identified on top by #FE/D=value) the expected running time (ERT) divided by the best ERT measured during BBOB-2009 for `texp` (x-coordinate) and for `def` (y-coordinate) and a summary across all five target values. A point on the upper triangle delimited by the diagonal indicates better performance for `texp`; a point on the lower right triangle better performance for `def`. A coordinate below 1 indicates that the algorithm's ERT is less than the best ERT measured during BBOB-2009. Given is also the number of times `texp` wins,loses, or draws with `def`. We marked with a + symbol those cases there is a statistically significant difference checked by two-sided Wilcoxon matched-pairs signed-rank test at the 0.05 α-level between the algorithms.

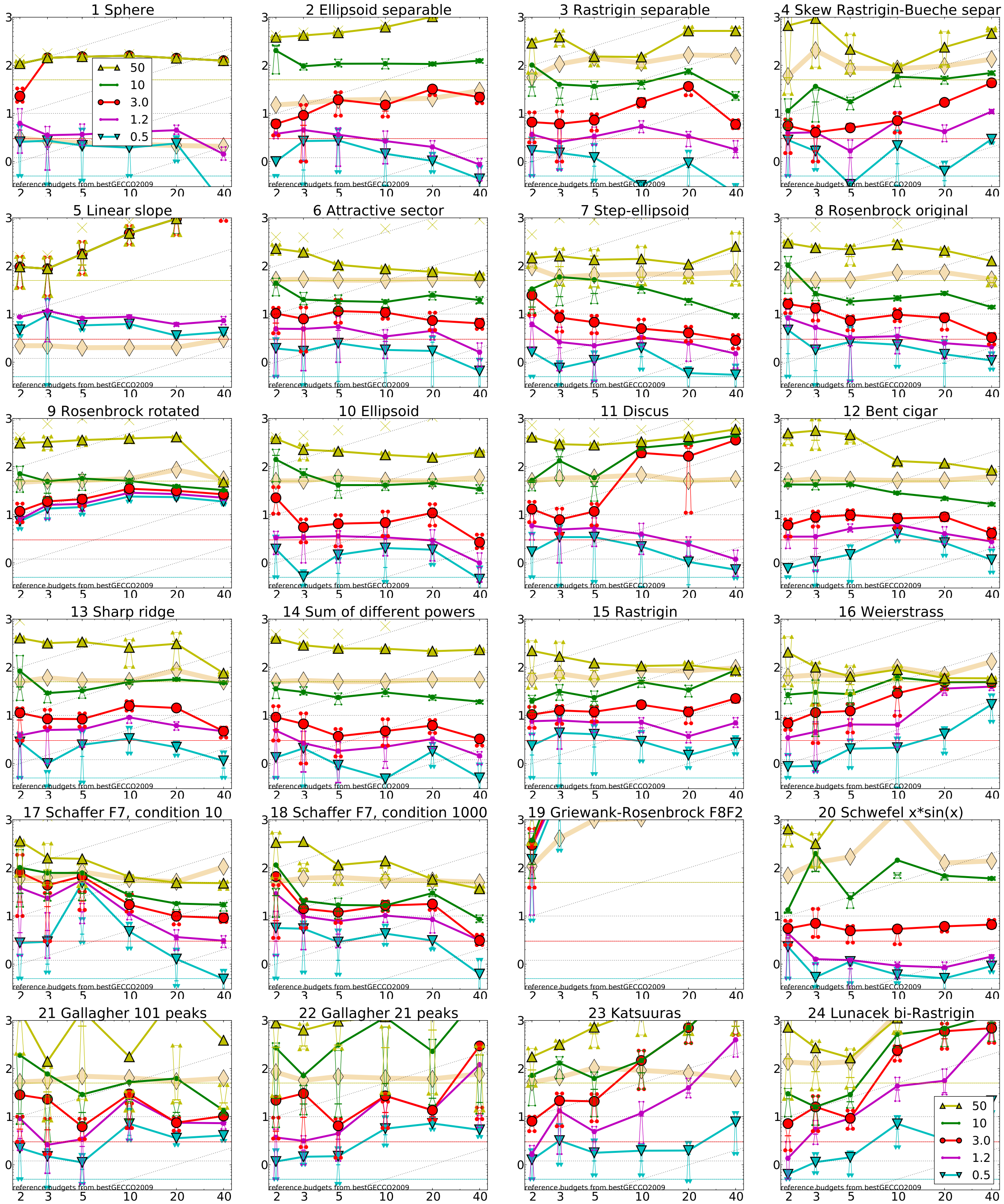


Figure 2: texp: Expected number of f-evaluations (ERT, lines) to reach $f_{\rm opt} + \Delta f$; median number of f-evaluations ($+$) to reach the most difficult target that was reached not always but at least once; maximum number of f-evaluations in any trial ($\times$); interquartile range with median (notched boxes) of simulated runlengths to reach $f_{\rm opt} + \Delta f$; all values are divided by dimension and plotted as $\log_{10}$ values versus dimension. Shown is the ERT for targets just not reached by the GECCO-BBOB-2009 best algorithm within the given budget kDIM, where k is shown in the legend. Numbers above ERT-symbols indicate the number of trials reaching the respective target. The light thick line with diamonds indicates the respective best result from BBOB-2009 for the most difficult target. Slanted grid lines indicate a scaling with $\mathcal{O}(\mathrm{DIM})$ compared to $\mathcal{O}(1)$ when using the respective 2009 best algorithm.

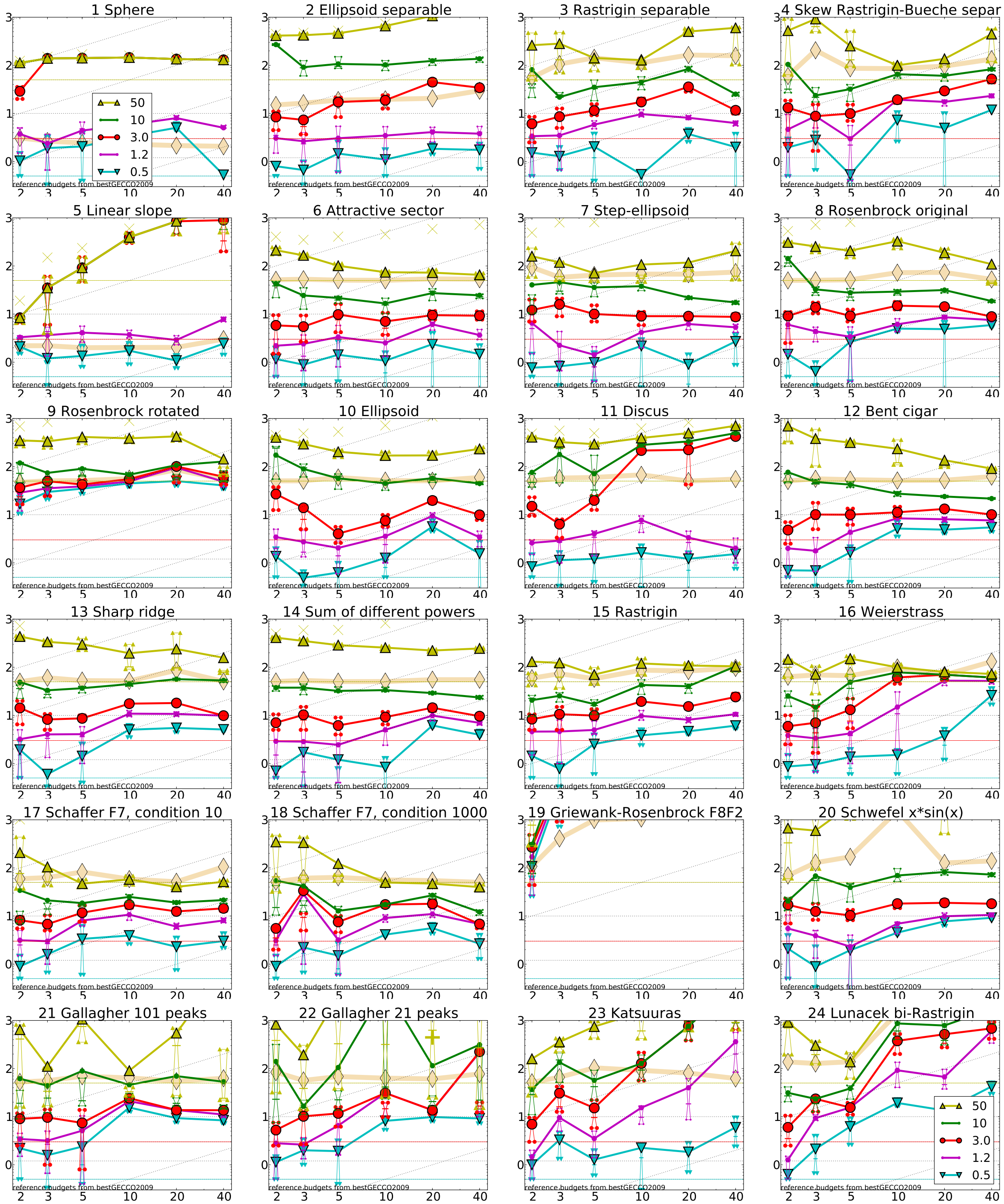


Figure 3: def:. Expected number of f-evaluations (ERT, lines) to reach $f_{\mathrm{opt}} + \Delta f$; median number of f-evaluations (+) to reach the most difficult target that was reached not always but at least once; maximum number of f-evaluations in any trial ($\times$); interquartile range with median (notched boxes) of simulated runlengths to reach $f_{\mathrm{opt}} + \Delta f$; all values are divided by dimension and plotted as $\log_{10}$ values versus dimension. Shown is the ERT for targets just not reached by the GECCO-BBOB-2009 best algorithm within the given budget kDIM, where k is shown in the legend. Numbers above ERT-symbols indicate the number of trials reaching the respective target. The light thick line with diamonds indicates the respective best result from BBOB-2009 for the most difficult target. Slanted grid lines indicate a scaling with $\mathcal{O}(\mathrm{DIM})$ compared to $\mathcal{O}(1)$ when using the respective 2009 best algorithm.

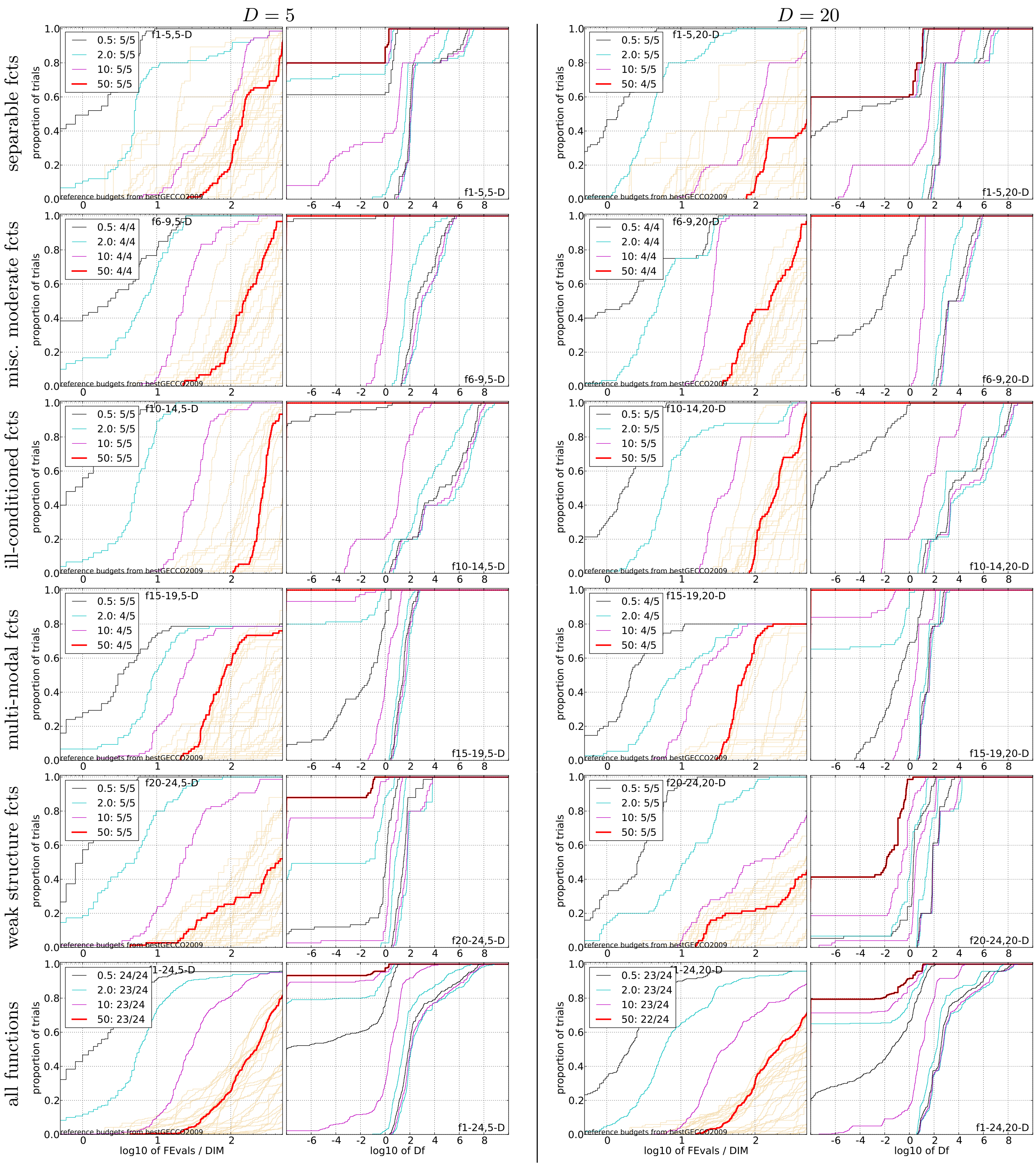


Figure 4: Empirical cumulative distribution functions (ECDF), plotting the fraction of trials with an outcome not larger than the respective value on the x-axis. Left subplots: ECDF of number of function evaluations (FEvals) divided by search space dimension D, to fall below $f_{\rm opt} + \Delta f$ where Δf is the target just not reached by the GECCO-BBOB-2009 best algorithm within a budget of $k \times \mathrm{DIM}$ evaluations, where k is the first value in the legend. Legends indicate for each target the number of functions that were solved in at least one trial within the displayed budget. Right subplots: ECDF of the best achieved Δf for running times of $0.5D, 1.2D, 3D, 10D, 100D, 1000D, \ldots$ function evaluations (from right to left cycling cyan-magenta-black...) and final Δf-value (red), where Δf and Df denote the difference to the optimal function value. Light brown lines in the background show ECDFs for the most difficult target of all algorithms benchmarked during BBOB-2009.

5-D

#FEs/D	0.5	1.2	3.0	10	50	#succ
f_1	*2.5e+1*:4.8	*1.6e+1*:7.6	*1.0e-8*:12	*1.0e-8*:12	*1.0e-8*:12	15/15
	2.3(3)	2.4(3)	62(5)	62(5)	62(5)	15/15
f_2	*1.6e+6*:2.9	*4.0e+5*:11	*4.0e+4*:15	*6.3e+2*:58	*1.0e-8*:95	15/15
	4.8(4)	1.6(1)	6.3(7)	9.4(3)	25(2)	15/15
f_3	*1.6e+2*:4.1	*1.0e+2*:15	*6.3e+1*:23	*2.5e+1*:73	*1.0e+1*:716	15/15
	1.5(1)	1.1(0.9)	1.6(0.9)	2.5(2)	1.1(1)	15/15
f_4	*2.5e+2*:2.6	*1.6e+2*:10	*1.0e+2*:19	*4.0e+1*:65	*1.6e+1*:434	15/15
	0.64(0.2)	0.86(1.0)	1.3(0.6)	1.3(0.6)	2.5(3)	15/15
f_5	*6.3e+1*:4.0	*4.0e+1*:10	*1.0e-8*:10	*1.0e-8*:10	*1.0e-8*:10	15/15
	7.3(2)	4.2(1)	89(71)	89(71)	89(71)	15/15
f_6	*1.0e+5*:3.0	*2.5e+4*:8.4	*1.0e+2*:16	*2.5e+1*:54	*2.5e-1*:254	15/15
	4.1(6)	3.2(4)	3.6(2)	1.7(1.0)	2.1(0.6)	15/15
f_7	*1.6e+2*:4.2	*1.0e+2*:6.2	*2.5e+1*:20	*4.0e+0*:54	*1.0e+0*:324	15/15
	1.3(1)	1.8(2)	1.7(1)	4.8(10)	2.1(2)	15/15
f_8	*1.0e+4*:4.6	*6.3e+3*:6.8	*1.0e+3*:18	*6.3e+1*:54	*1.6e+0*:258	15/15
	2.9(3)	2.4(2)	2.1(1)	1.7(0.5)	4.3(3)	15/15
f_9	*2.5e+1*:20	*1.6e+1*:26	*1.0e+1*:35	*4.0e+0*:62	*1.6e-2*:256	15/15
	3.6(2)	3.3(1)	3.0(2)	4.6(4)	7.0(2)	15/15
f_{10}	*2.5e+6*:2.9	*6.3e+5*:7.0	*2.5e+5*:17	*6.3e+3*:54	*2.5e+1*:297	15/15
	2.6(4)	2.6(3)	1.9(2)	3.9(3)	3.5(1)	15/15
f_{11}	*1.0e+6*:3.0	*6.3e+4*:6.2	*6.3e+2*:16	*6.3e+1*:74	*6.3e-1*:298	15/15
	5.7(6)	4.3(3)	3.7(2)	4.0(6)	4.7(0.8)	15/15
f_{12}	*4.0e+7*:3.6	*1.6e+7*:7.6	*4.0e+6*:19	*1.6e+4*:52	*1.0e+0*:268	15/15
	2.1(2)	3.4(2)	2.6(1)	4.2(0.6)	8.7(9)	15/15
f_{13}	*1.0e+3*:2.8	*6.3e+2*:8.4	*4.0e+2*:17	*6.3e+1*:52	*6.3e-2*:264	15/15
	4.4(5)	3.0(3)	2.5(2)	3.1(1)	6.4(2)	15/15
f_{14}	*1.6e+1*:3.0	*1.0e+1*:10	*6.3e+0*:15	*2.5e-1*:53	*1.0e-5*:251	15/15
	1.5(2)	0.93(1)	1.2(0.9)	2.2(0.7)	4.9(0.7)	15/15
f_{15}	*1.6e+2*:3.0	*1.0e+2*:13	*6.3e+1*:24	*4.0e+1*:55	*1.6e+1*:289	5/5
	6.8(8)	2.8(2)	2.4(1)	2.1(2)	2.1(3)	15/15
f_{16}	*4.0e+1*:4.8	*2.5e+1*:16	*1.6e+1*:46	*1.0e+1*:120	*4.0e+0*:334	15/15
	2.1(2)	2.0(2)	1.3(1)	1.1(1)	0.96(1.0)	15/15
f_{17}	*1.0e+1*:5.2	*6.3e+0*:26	*4.0e+0*:57	*2.5e+0*:110	*6.3e-1*:412	15/15
	47(7)	11(2)	5.9(2)	3.6(3)	1.9(4)	15/15
f_{18}	*6.3e+1*:3.4	*4.0e+1*:7.2	*2.5e+1*:20	*1.6e+1*:58	*1.6e+0*:318	15/15
	4.2(5)	5.4(4)	3.0(2)	1.4(0.8)	1.8(0.7)	15/15
f_{19}	*1.6e-1*:172	*1.0e-1*:242	*6.3e-2*:675	*4.0e-2*:3078	*2.5e-2*:4946	15/15
	895(958)	746(669)	284(240)	68(52)	51(44)	15/15
f_{20}	*6.3e+3*:5.1	*4.0e+3*:8.4	*4.0e+1*:15	*2.5e+0*:69	*1.0e+0*:851	15/15
	1.1(1)	0.72(0.8)	1.6(0.6)	1.7(1)	17(19)	15/15
f_{21}	*4.0e+1*:3.9	*2.5e+1*:11	*1.6e+1*:31	*6.3e+0*:73	*1.6e+0*:347	5/5
	1.4(1)	1.6(2)	1.0(0.9)	2.0(1)	23(43)	15/15
f_{22}	*6.3e+1*:3.6	*4.0e+1*:15	*2.5e+1*:32	*1.0e+1*:71	*1.6e+0*:341	5/5
	2.1(2)	1.5(1)	1.0(1)	22(9)	14(27)	15/15
f_{23}	*1.0e+1*:3.0	*6.3e+0*:9.0	*4.0e+0*:33	*2.5e+0*:84	*1.0e+0*:518	15/15
	3.0(3)	2.7(4)	3.1(3)	3.7(4)	7.0(7)	15/15
f_{24}	*6.3e+1*:15	*4.0e+1*:37	*4.0e+1*:37	*2.5e+1*:118	*1.6e+1*:692	15/15
	0.49(0.4)	1.3(1)	1.3(1)	1.2(1.0)	1.2(1)	15/15

20-D

#FEs/D	0.5	1.2	3.0	10	50	#succ
f_1	*6.3e+1*:24	*4.0e+1*:42	*1.0e-8*:43	*1.0e-8*:43	*1.0e-8*:43	15/15
	2.0(1)	2.1(0.8)	66(4)	66(4)	66(4)	15/15
f_2	*4.0e+6*:29	*2.5e+6*:42	*1.0e+5*:65	*1.0e+4*:207	*1.0e-8*:412	15/15
	0.72(0.6)	0.96(0.5)	10(5)	10(2)	50(2)	15/15
f_3	*6.3e+2*:33	*4.0e+2*:44	*1.6e+2*:109	*1.0e+2*:255	*2.5e+1*:3277	15/15
	0.57(0.5)	1.5(0.8)	6.7(4)	5.9(1)	3.2(2)	15/15
f_4	*6.3e+2*:22	*4.0e+2*:91	*2.5e+2*:250	*1.6e+2*:332	*6.3e+1*:1927	15/15
	0.59(0.7)	0.92(0.5)	1.4(0.4)	3.2(1)	2.5(2)	15/15
f_5	*2.5e+2*:19	*1.6e+2*:34	*1.0e-8*:41	*1.0e-8*:41	*1.0e-8*:41	15/15
	3.8(0.6)	3.6(0.9)	471(320)	471(320)	471(320)	15/15
f_6	*2.5e+5*:16	*6.3e+4*:43	*1.6e+4*:62	*1.6e+2*:353	*1.6e+1*:1078	15/15
	2.1(2)	2.1(2)	2.4(1)	1.4(0.5)	1.4(0.3)	15/15
f_7	*1.0e+3*:11	*4.0e+2*:39	*2.5e+2*:74	*6.3e+1*:319	*1.0e+1*:1351	15/15
	1.1(2)	1.3(0.9)	1.1(0.6)	1.2(0.3)	1.6(2)	15/15
f_8	*4.0e+4*:19	*2.5e+4*:35	*4.0e+3*:67	*2.5e+2*:231	*1.6e+1*:1470	15/15
	1.6(2)	1.4(1)	2.5(2)	2.3(0.3)	2.9(1.0)	15/15
f_9	*1.0e+2*:357	*6.3e+1*:560	*4.0e+1*:684	*2.5e+1*:756	*1.0e+1*:1716	15/15
	1.3(0.2)	0.96(0.1)	0.92(0.1)	1.0(0.1)	4.8(1)	15/15
f_{10}	*1.6e+6*:15	*1.0e+6*:27	*4.0e+5*:70	*6.3e+4*:231	*4.0e+3*:1015	15/15
	2.5(2)	2.1(2)	3.1(2)	4.0(1)	3.1(0.7)	15/15
f_{11}	*4.0e+4*:11	*2.5e+3*:27	*1.6e+2*:313	*1.0e+2*:481	*1.0e+1*:1002	15/15
	1.9(2)	1.8(1)	11(9)	13(1)	8.4(0.6)	15/15
f_{12}	*1.0e+8*:23	*6.3e+7*:39	*2.5e+7*:76	*4.0e+6*:209	*1.0e+1*:1042	15/15
	2.3(3)	2.0(2)	2.4(0.8)	2.1(0.4)	2.3(0.2)	15/15
f_{13}	*1.6e+3*:28	*1.0e+3*:64	*6.3e+2*:79	*4.0e+1*:211	*2.5e+0*:1724	15/15
	1.6(0.8)	1.9(0.4)	3.6(0.4)	5.3(0.6)	3.6(3)	15/15
f_{14}	*2.5e+1*:15	*1.6e+1*:42	*1.0e+1*:75	*1.6e+0*:219	*6.3e-4*:1106	15/15
	2.5(2)	1.5(0.8)	1.6(0.8)	2.2(0.3)	3.9(0.4)	15/15
f_{15}	*6.3e+2*:15	*4.0e+2*:67	*2.5e+2*:292	*1.6e+2*:846	*1.0e+2*:1671	15/15
	1.9(1)	1.1(0.5)	0.80(0.3)	0.80(0.3)	1.3(0.3)	15/15
f_{16}	*4.0e+1*:26	*2.5e+1*:127	*1.6e+1*:540	*1.6e+1*:540	*1.0e+1*:1384	15/15
	3.1(4)	5.7(3)	1.8(0.7)	1.8(0.7)	0.86(0.3)	15/15
f_{17}	*1.6e+1*:11	*1.0e+1*:63	*6.3e+0*:305	*4.0e+0*:468	*1.0e+0*:1030	15/15
	2.4(4)	1.2(0.9)	**0.65**(0.2)$^{\downarrow}$	0.78(0.4)	0.96(0.2)	15/15
f_{18}	*4.0e+1*:116	*2.5e+1*:252	*1.6e+1*:430	*1.0e+1*:621	*4.0e+0*:1090	15/15
	0.53(0.3)	0.67(0.4)	0.83(0.3)	0.97(0.2)	1.1(0.5)	15/15
f_{19}	*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
	2.6(2)	2.0(2)	2.6(2)	4.0(3)	5.5(2)	15/15
f_{20}	*1.6e+4*:38	*1.0e+4*:42	*2.5e+2*:62	*2.5e+0*:250	*1.6e+0*:2536	15/15
	0.26(0.2)$^{\downarrow}$	**0.41**(0.2)$^{\downarrow 4}$	2.0(0.8)	5.5(2)	41(58)	15/15
f_{21}	*6.3e+1*:36	*4.0e+1*:77	*4.0e+1*:77	*1.6e+1*:456	*4.0e+0*:1094	15/15
	2.0(0.8)	2.0(1)	2.0(1)	2.7(0.7)	66(162)	15/15
f_{22}	*6.3e+1*:45	*4.0e+1*:68	*4.0e+1*:68	*1.6e+1*:231	*6.3e+0*:1219	15/15
	3.2(2)	4.0(3)	4.0(3)	20(27)	194(423)	15/15
f_{23}	*6.3e+0*:29	*4.0e+0*:118	*2.5e+0*:306	*2.5e+0*:306	*1.0e+0*:1614	15/15
	1.4(1.0)	6.8(5)	46(61)	46(61)	28(36)	15/15
f_{24}	*2.5e+2*:208	*1.6e+2*:918	*1.0e+2*:6628	*6.3e+1*:9885	*4.0e+1*:31629	15/15
	0.33(0.3)$^{\downarrow 3}$	1.2(1)	1.8(2)	1.4(1)	1.2(1)	15/15

Table 2: `texp`: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009. The ERT and in braces, as dispersion measure, the half difference between 90 and 10%-tile of bootstrapped run lengths appear in the second row of each cell, the best ERT (preceded by the target Δf-value in *italics*) in the first. #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. Bold entries are statistically significantly better (according to the rank-sum test) compared to the best algorithm in BBOB-2009, with $p = 0.05$ or $p = 10^{-k}$ when the number $k > 1$ is following the $\downarrow$ symbol, with Bonferroni correction by the number of functions.

Proceedings of the Workshop on Scaling Behaviours of Landscapes, Parameters and Algorithms of the Genetic and Evolutionary Computation Conference, GECCO'11, pages 703–706, New York, NY, USA, 2011. ACM.

[7] T. Liao, M. A. Montes de Oca, and T. Stützle. Computational results for an automatically tuned CMA-ES with increasing population size on the CEC'05 benchmark set. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 2013. In press.

[8] T. Liao and T. Stützle. Bounding the population size of IPOP-CMA-ES on the noiseless BBOB testbed. In *Proceedings of Genetic and Evolutionary Computation Conference*, GECCO Companion'13, 2013. submitted.

[9] T. Liao and T. Stützle. Testing the impact of parameter tuning on a variant of IPOP-CMA-ES with a bounded maximum population size on the noiseless BBOB testbed. In *Proceedings of Genetic and Evolutionary Computation Conference*, GECCO Companion'13, 2013. submitted.

[10] M. López-Ibáñez, J. Dubois-Lacoste, T. Stützle, and M. Birattari. The `irace` package, iterated race for automatic algorithm configuration. Technical Report TR/IRIDIA/2011-004, IRIDIA, Université Libre de Bruxelles, Belgium, 2011.

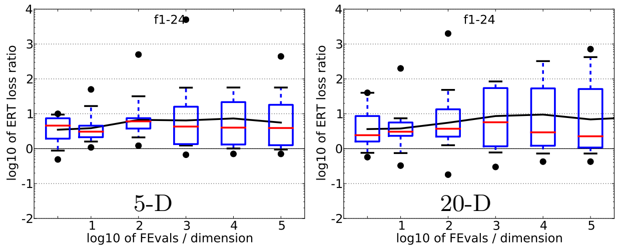
5-D

#FEs/D	0.5	1.2	3.0	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
	2.1(2)	2.9(2)	58(5)	58(5)	58(5)	15/15
f2	*1.6e+6*:2.9	*4.0e+5*:11	*4.0e+4*:15	*6.3e+2*:58	*1.0e-8*:95	15/15
	2.6(2)	1.4(1)	5.7(4)	9.3(5)	24(3)	15/15
f3	*1.6e+2*:4.1	*1.0e+2*:15	*6.3e+1*:23	*2.5e+1*:73	*1.0e+1*:716	15/15
	2.5(3)	2.0(1)	2.5(1)	2.4(2)	0.99(1)	15/15
f4	*2.5e+2*:2.6	*1.6e+2*:10	*1.0e+2*:19	*4.0e+1*:65	*1.6e+1*:434	15/15
	1.0(0.2)	1.5(2)	2.6(2)	2.4(1.0)	2.9(3)	15/15
f5	*6.3e+1*:4.0	*4.0e+1*:10	*1.0e-8*:10	*1.0e-8*:10	*1.0e-8*:10	15/15
	1.7(1)	2.1(1)	46(43)	46(43)	46(43)	15/15
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
	2.4(2)	2.0(2)	3.1(3)	2.0(0.9)	2.0(0.3)	15/15
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
	1.2(1)	1.2(1)	2.5(2)	3.3(1)	1.1(0.4)	15/15
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
	2.9(3)	2.5(2)	2.6(1)	2.6(0.9)	4.0(1)	15/15
f9	*2.5e+1*:20	*1.6e+1*:26	*1.0e+1*:35	*4.0e+0*:62	*1.6e-2*:256	15/15
	8.8(4)	7.5(3)	6.1(2)	7.3(5)	8.1(5)	15/15
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
	1.1(0.9)	1.5(2)	1.2(0.8)	5.3(3)	3.4(0.9)	15/15
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
	2.0(2)	3.2(4)	6.2(3)	4.8(6)	4.9(0.4)	15/15
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
	2.3(2)	2.9(2)	2.6(1)	4.1(1)	5.9(3)	15/15
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
	2.5(2)	2.4(2)	2.6(0.7)	3.5(1)	5.7(3)	15/15
f14	*1.6e+1*:3.0	*1.0e+1*:10	*6.3e+0*:15	*2.5e-1*:53	*1.0e-5*:251	15/15
	2.0(2)	1.2(1)	2.0(2)	3.1(1)	5.7(0.4)	15/15
f15	*1.6e+2*:3.0	*1.0e+2*:13	*6.3e+1*:24	*4.0e+1*:55	*1.6e+1*:289	5/5
	4.2(4)	1.9(2)	2.0(1)	1.5(0.7)	1.2(0.6)	15/15
f16	*4.0e+1*:4.8	*2.5e+1*:16	*1.6e+1*:46	*1.0e+1*:120	*4.0e+0*:334	15/15
	1.4(1)	1.3(1)	1.4(1)	2.0(2)	2.3(1)	15/15
f17	*1.0e+1*:5.2	*6.3e+0*:26	*4.0e+0*:57	*2.5e+0*:110	*6.3e-1*:412	15/15
	3.2(3)	1.5(1)	1.0(0.5)	0.84(0.3)	**0.57**(0.2)$^{\downarrow 3}$	15/15
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
	2.2(2)	2.2(2)	1.9(2)	1.1(0.6)	1.9(0.4)	15/15
f19	*1.6e-1*:172	*1.0e-1*:242	*6.3e-2*:675	*4.0e-2*:3078	*2.5e-2*:4946	15/15
	614(657)	519(658)	263(299)	72(77)	46(48)	15/15
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
	1.9(2)	1.4(2)	3.4(1)	2.9(2)	12(11)	15/15
f21	*4.0e+1*:3.9	*2.5e+1*:11	*1.6e+1*:31	*6.3e+0*:73	*1.6e+0*:347	5/5
	3.0(3)	2.4(3)	1.2(1)	6.1(11)	15(11)	15/15
f22	*6.3e+1*:3.6	*4.0e+1*:15	*2.5e+1*:32	*1.0e+1*:71	*1.6e+0*:341	5/5
	2.7(2)	2.2(2)	1.8(1)	7.4(11)	47(60)	15/15
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
	2.1(2)	1.9(2)	2.3(2)	3.4(3)	7.2(5)	15/15
f24	*6.3e+1*:15	*4.0e+1*:37	*4.0e+1*:37	*2.5e+1*:118	*1.6e+1*:692	15/15
	2.1(2)	2.1(0.7)	2.1(0.7)	1.6(1)	0.99(0.6)	15/15

20-D

#FEs/D	0.5	1.2	3.0	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
	4.2(2)	3.8(0.7)	63(5)	63(5)	63(5)	15/15
f2	*4.0e+6*:29	*2.5e+6*:42	*1.0e+5*:65	*1.0e+4*:207	*1.0e-8*:412	15/15
	1.3(1)	2.0(1)	14(3)	12(3)	52(2)	15/15
f3	*6.3e+2*:33	*4.0e+2*:44	*1.6e+2*:109	*1.0e+2*:255	*2.5e+1*:3277	15/15
	2.3(1)	3.7(0.8)	6.5(3)	6.6(2)	3.1(2)	15/15
f4	*6.3e+2*:22	*4.0e+2*:91	*2.5e+2*:250	*1.6e+2*:332	*6.3e+1*:1927	15/15
	4.6(6)	3.8(0.9)	2.3(0.5)	3.7(1)	1.4(0.3)	15/15
f5	*2.5e+2*:19	*1.6e+2*:34	*1.0e-8*:41	*1.0e-8*:41	*1.0e-8*:41	15/15
	1.1(0.7)	1.7(1.0)	418(295)	418(295)	418(295)	15/15
f6	*2.5e+5*:16	*6.3e+4*:43	*1.6e+4*:62	*1.6e+2*:353	*1.6e+1*:1078	15/15
	3.0(3)	2.8(1)	3.1(1)	1.5(0.6)	1.3(0.3)	15/15
f7	*1.0e+3*:11	*4.0e+2*:39	*2.5e+2*:74	*6.3e+1*:319	*1.0e+1*:1351	15/15
	1.7(2)	3.2(0.7)	2.5(0.6)	1.4(0.3)	1.7(2)	15/15
f8	*4.0e+4*:19	*2.5e+4*:35	*4.0e+3*:67	*2.5e+2*:231	*1.6e+1*:1470	15/15
	5.2(4)	4.9(0.6)	4.2(0.4)	2.7(0.6)	2.6(0.8)	15/15
f9	*1.0e+2*:357	*6.3e+1*:560	*4.0e+1*:684	*2.5e+1*:756	*1.0e+1*:1716	15/15
	2.8(0.7)	3.4(5)	2.9(5)	2.8(4)	5.0(1)	15/15
f10	*1.6e+6*:15	*1.0e+6*:27	*4.0e+5*:70	*6.3e+4*:231	*4.0e+3*:1015	15/15
	7.5(4)	6.9(2)	5.6(1)	5.0(2)	3.4(0.9)	15/15
f11	*4.0e+4*:11	*2.5e+3*:27	*1.6e+2*:313	*1.0e+2*:481	*1.0e+1*:1002	15/15
	2.2(3)	2.5(3)	14(9)	14(3)	10(0.6)	15/15
f12	*1.0e+8*:23	*6.3e+7*:39	*2.5e+7*:76	*4.0e+6*:209	*1.0e+1*:1042	15/15
	4.2(3)	4.1(1)	3.5(0.4)	2.3(0.2)	2.6(2)	15/15
f13	*1.6e+3*:28	*1.0e+3*:64	*6.3e+2*:79	*4.0e+1*:211	*2.5e+0*:1724	15/15
	4.0(1)	3.3(0.6)	4.6(0.8)	5.4(0.3)	2.8(3)	15/15
f14	*2.5e+1*:15	*1.6e+1*:42	*1.0e+1*:75	*1.6e+0*:219	*6.3e-4*:1106	15/15
	8.5(3)	4.7(2)	3.8(0.7)	2.7(0.4)	4.0(0.5)	15/15
f15	*6.3e+2*:15	*4.0e+2*:67	*2.5e+2*:292	*1.6e+2*:846	*1.0e+2*:1671	15/15
	6.1(2)	2.4(0.5)	1.0(0.2)	0.95(0.5)	1.3(0.2)	15/15
f16	*4.0e+1*:26	*2.5e+1*:127	*1.6e+1*:540	*1.6e+1*:540	*1.0e+1*:1384	15/15
	2.9(3)	8.3(3)	2.6(0.7)	2.6(0.7)	1.2(0.3)	15/15
f17	*1.6e+1*:11	*1.0e+1*:63	*6.3e+0*:305	*4.0e+0*:468	*1.0e+0*:1030	15/15
	4.3(4)	1.9(0.8)	0.82(0.2)	0.81(0.1)	**0.78**(0.1)$^{\downarrow}$	15/15
f18	*4.0e+1*:116	*2.5e+1*:252	*1.6e+1*:430	*1.0e+1*:621	*4.0e+0*:1090	15/15
	0.97(0.3)	0.87(0.4)	0.84(0.2)	0.87(0.2)	0.87(0.2)	15/15
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
	5.1(5)	4.0(3)	5.0(6)	6.6(6)	7.5(5)	15/15
f20	*1.6e+4*:38	*1.0e+4*:42	*2.5e+2*:62	*2.5e+0*:250	*1.6e+0*:2536	15/15
	4.0(1)	4.7(1)	6.0(1)	6.6(2)	31(30)	15/15
f21	*6.3e+1*:36	*4.0e+1*:77	*4.0e+1*:77	*1.6e+1*:456	*4.0e+0*:1094	15/15
	5.2(1)	3.6(0.6)	3.6(0.6)	3.0(6)	10(11)	15/15
f22	*6.3e+1*:45	*4.0e+1*:68	*4.0e+1*:68	*1.6e+1*:231	*6.3e+0*:1219	15/15
	4.4(1)	4.0(1)	4.0(1)	10(26)	1610(1374)	14/15
f23	*6.3e+0*:29	*4.0e+0*:118	*2.5e+0*:306	*2.5e+0*:306	*1.0e+0*:1614	15/15
	1.3(1)	6.8(7)	50(36)	50(36)	35(39)	15/15
f24	*2.5e+2*:208	*1.6e+2*:918	*1.0e+2*:6628	*6.3e+1*:9885	*4.0e+1*:31629	15/15
	1.3(0.3)	1.5(1)	1.6(2)	1.6(1)	1.4(0.8)	15/15

Table 3: `def`: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009. The ERT and in braces, as dispersion measure, the half difference between 90 and 10%-tile of bootstrapped run lengths appear in the second row of each cell, the best ERT (preceded by the target Δf-value in *italics*) in the first. #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. Bold entries are statistically significantly better (according to the rank-sum test) compared to the best algorithm in BBOB-2009, with $p = 0.05$ or $p = 10^{-k}$ when the number $k > 1$ is following the $\downarrow$ symbol, with Bonferroni correction by the number of functions.



#FEs/D	best	10%	25%	**med**	75%	90%
	***f*1–*f*24 in 5-D**, maxFE/D=1.00e6					
2	0.49	0.84	1.8	4.6	7.7	10
10	1.1	1.5	2.1	3.1	4.7	24
100	1.2	2.0	3.8	6.1	7.7	42
1e3	0.67	1.2	1.3	4.3	19	65
1e4	0.71	0.97	1.3	4.0	23	65
1e5	0.71	0.92	1.2	3.9	19	65
1e6	0.50	0.79	1.2	3.9	23	65
$\mathrm{RL_{US}}$/D	1e6	1e6	1e6	1e6	1e6	1e6

#FEs/D	best	10%	25%	**med**	75%	90%
	***f*1–*f*24 in 20-D**, maxFE/D=1.00e6					
2	0.57	0.71	1.5	2.4	10	40
10	0.33	0.67	2.2	3.1	5.8	27
100	0.18	1.2	2.2	3.7	15	50
1e3	0.30	0.73	1.2	5.7	58	1.3e2
1e4	0.43	0.65	1.2	2.9	57	4.3e2
1e5	0.43	0.65	1.0	2.3	53	4.8e2
1e6	0.43	0.65	1.0	2.3	61	6.9e2
1e7	0.43	0.65	1.0	4.0	65	3.1e3
$\mathrm{RL_{US}}$/D	1e6	1e6	1e6	1e6	1e6	1e6

Figure 5: ERT loss ratio versus the budget (both in number of f-evaluations divided by dimension). The target value f_t for a given budget FEvals is the best target f-value reached within the budget by the given algorithm. Shown is the ERT of the given algorithm divided by best ERT seen in GECCO-BBOB-2009 for the target f_t, or, if the best algorithm reached a better target within the budget, the budget divided by the best ERT. Line: geometric mean. Box-Whisker error bar: 25-75%-ile with median (box), 10-90%-ile (caps), and minimum and maximum ERT loss ratio (points). The vertical line gives the maximal number of function evaluations in a single trial in this function subset. See also Figure 6 for results on each function subgroup.

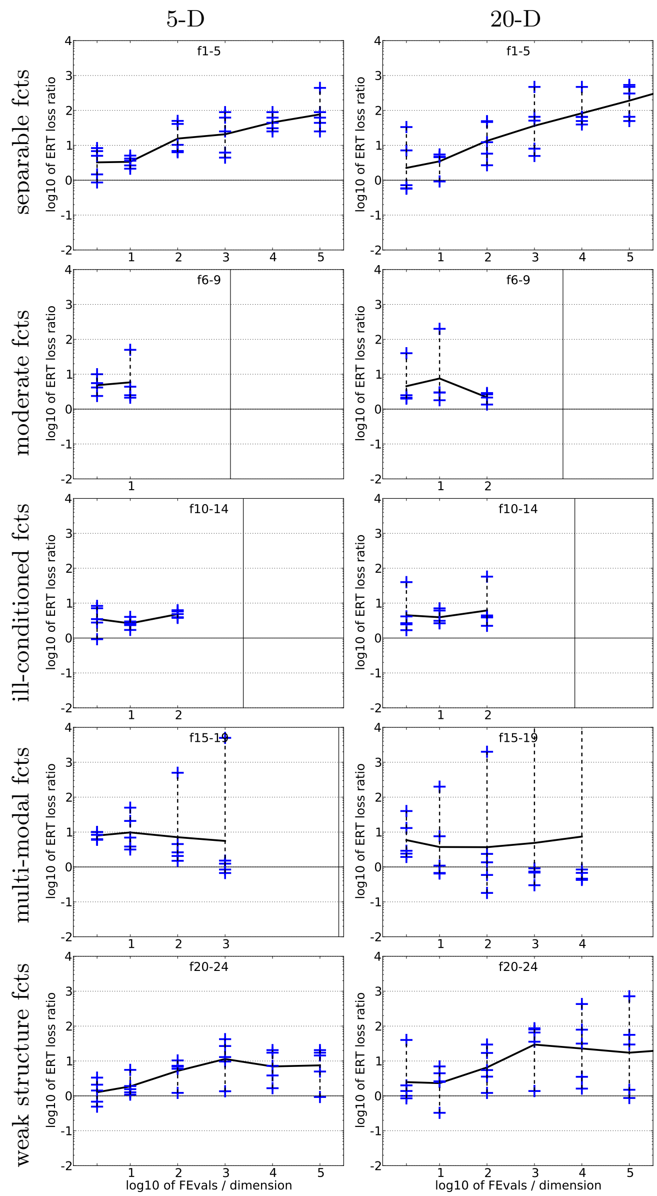


Figure 6: ERT loss ratios (see Figure 5 for details). Each cross (+) represents a single function, the line is the geometric mean.