Black-Box Optimization Benchmarking Template for Noiseless Function Testbed

Nicholas Robinson

Sam McNaughton

ABSTRACT

Comparison of Neural Network, trained using Genetic Algorithm, with COCOs given noiseless functions.

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

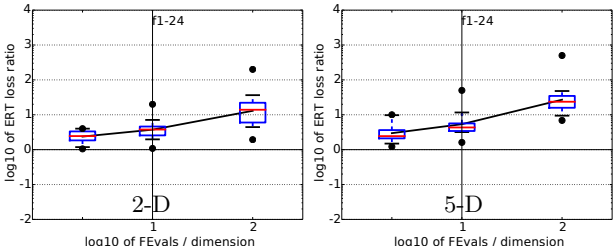
Algorithms

Keywords

Benchmarking, Black-box optimization

1. RESULTS

Results of NNCompare from experiments on the benchmark functions are presented in Figures 1, 2, 3, and 4 and in Tables 1.



	f_1–f_{24} in 2-D, maxFE/D=10					
#FEs/D	best	10%	25%	**med**	75%	90%
2	1.1	1.1	1.8	2.5	3.3	4.0
10	1.1	1.9	2.6	3.8	4.7	8.9
100	1.9	4.2	5.8	14	23	38
RL_{US}/D	10	10	10	10	10	10
	f_1–f_{24} in 5-D, maxFE/D=10					
#FEs/D	best	10%	25%	**med**	75%	90%
2	1.2	1.4	2.1	2.4	3.7	10
10	1.6	3.1	3.4	4.3	5.8	16
100	6.9	9.0	16	24	35	96
RL_{US}/D	10	10	10	10	10	10

Figure 3: ERT loss ratio versus the budget in number of f-evaluations divided by dimension. For each given budget FEvals, the target value f_t is computed as the best target f-value reached within the budget by the given algorithm. Shown is then the ERT to reach f_t for the given algorithm or the budget, if the GECCO-BBOB-2009 best algorithm reached a better target within the budget, divided by the best ERT seen in GECCO-BBOB-2009 to reach f_t. 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 4 for results on each function subgroup.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
GECCO'13, July 6-10, 2013, Amsterdam, The Netherlands.
Copyright 2013 ACM TBA ...$10.00.

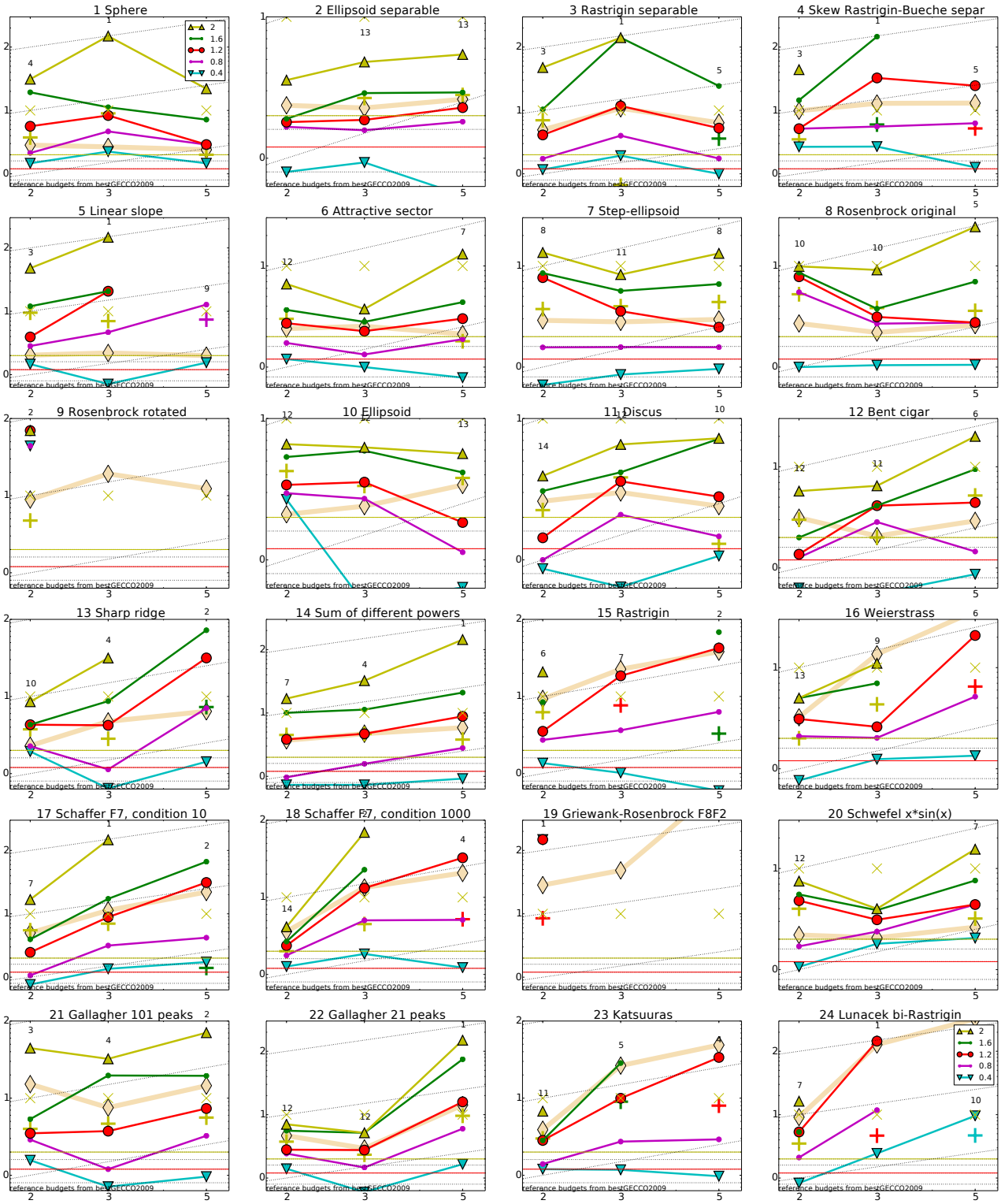


Figure 1: Expected number of f-evaluations (ERT, lines) to reach $f_{\mathrm{opt}} + \Delta f$; median number of f-evaluations (+) to reach the most difficult target that was reached not always but at least once; maximum number of f-evaluations in any trial (×); all values are divided by dimension and plotted as $\log_{10}$ values versus dimension. Shown is the ERT for targets just not reached by the artificial GECCO-BBOB-2009 best algorithm within the given budget $k \times \mathrm{DIM}$, 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.

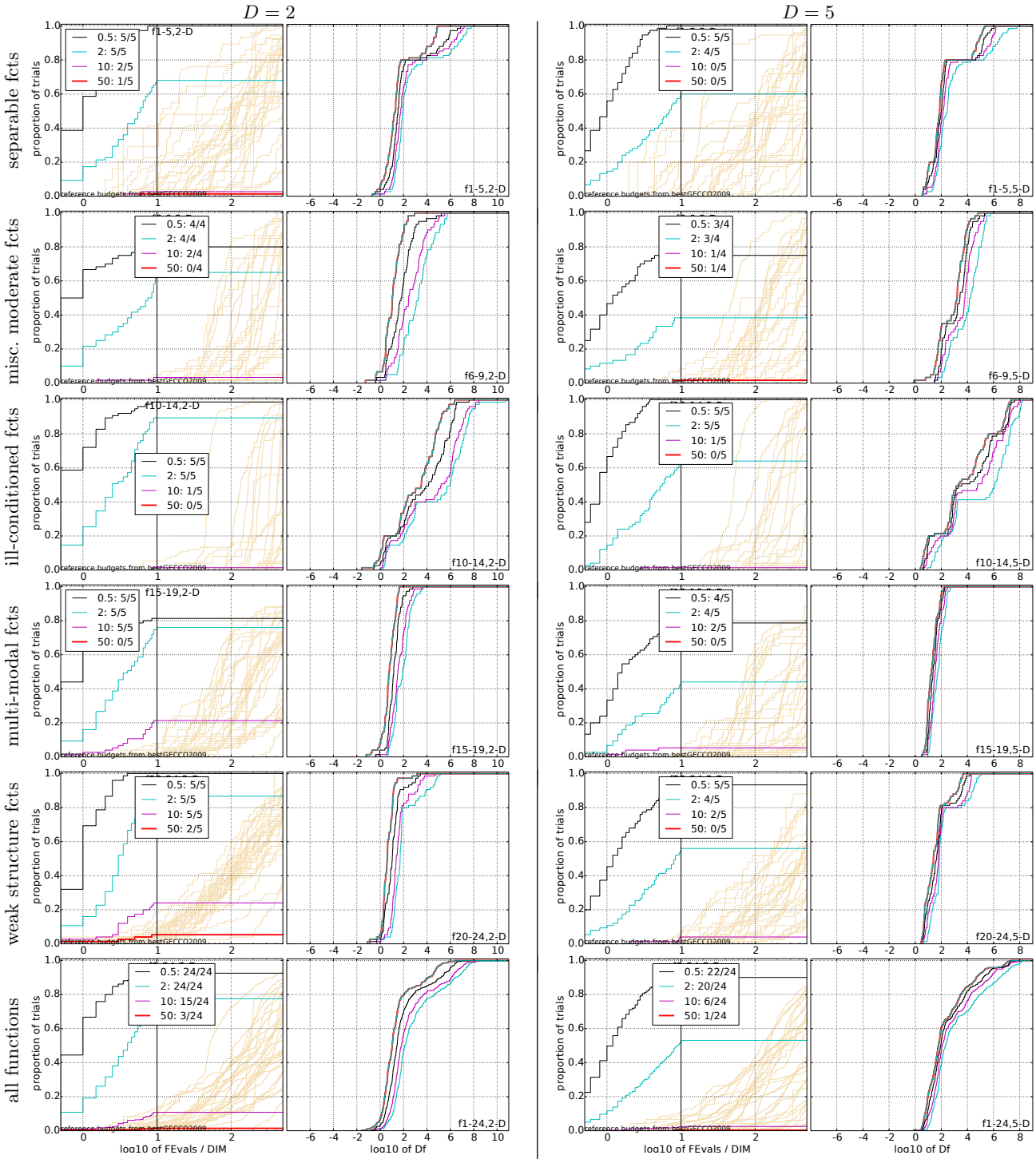


Figure 2: Empirical cumulative distribution functions (ECDF), plotting the fraction of trials with an outcome not larger than the respective value on the x-axis. Left subplots: ECDF of number of function evaluations (FEvals) divided by search space dimension D, to fall below $f_{\mathrm{opt}} + \Delta f$ 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.

2-D

#FEs/D	0.5	1.2	3	10	50	#succ
$\mathbf{f_1}$	*1.6e+1*:1.2	*4.0e+0*:2.6	*2.5e-2*:6.2	*1.0e-8*:6.2	*1.0e-8*:6.2	15/15
	2.4(2)	4.3(5)	∞	∞	∞*20*	0/15
$\mathbf{f_2}$	*1.0e+7*:1.4	*1.6e+6*:2.7	*1.0e+5*:6.1	*6.3e-1*:20	*1.0e-8*:30	15/15
	1.1(0.7)	1.3(1)	1.6(1)	∞	∞*20*	0/15
$\mathbf{f_3}$	*1.0e+2*:1.4	*4.0e+1*:4.1	*2.5e+1*:6.6	*6.3e+0*:26	*2.5e+0*:112	15/15
	1.7(1)	2.0(2)	3.1(4)	11(12)	2.6(3)	1/15
$\mathbf{f_4}$	*6.3e+1*:2.4	*4.0e+1*:5.2	*2.5e+1*:8.5	*1.0e+1*:22	*2.5e+0*:120	5/5
	2.2(3)	2.0(2)	3.4(4)	13(13)	∞*20*	0/15
$\mathbf{f_5}$	*4.0e+1*:1.4	*1.6e+1*:3.5	*1.0e-8*:4.4	*1.0e-8*:4.4	*1.0e-8*:4.4	15/15
	2.1(2)	2.3(2)	∞	∞	∞*20*	0/15
$\mathbf{f_6}$	*6.3e+4*:1.4	*1.0e+2*:2.8	*1.6e+1*:10	*1.0e+0*:23	*2.5e-6*:103	15/15
	1.7(3)	1.9(2)	1.6(2)	∞	∞*20*	0/15
$\mathbf{f_7}$	*4.0e+2*:1.6	*1.0e+1*:3.2	*2.5e+0*:14	*1.6e+0*:21	*1.6e-2*:188	15/15
	0.83(0.3)	4.8(4)	4.8(5)	14(16)	∞*20*	0/15
$\mathbf{f_8}$	*2.5e+3*:1.2	*1.0e+2*:3.2	*6.3e+0*:7.0	*1.6e-1*:27	*1.6e-6*:100	15/15
	1.7(2)	3.4(3)	6.9(8)	∞	∞*20*	0/15
$\mathbf{f_9}$	*6.3e+0*:13	*4.0e+0*:15	*2.5e+0*:15	*2.5e-1*:21	*1.0e-8*:94	15/15
	7.0(8)	6.0(6)	9.4(10)	14(15)	∞*20*	0/15
$\mathbf{f_{10}}$	*1.6e+6*:2.0	*4.0e+5*:3.2	*6.3e+2*:8.8	*1.0e+1*:30	*2.5e-8*:101	15/15
	2.7(4)	2.1(2)	33(38)	∞	∞*20*	0/15
$\mathbf{f_{11}}$	*1.0e+7*:1.1	*1.6e+6*:3.2	*1.0e+4*:6.6	*4.0e+1*:23	*4.0e-8*:100	15/15
	1.6(0.9)	0.90(0.6)	4.5(4)	13(14)	∞*20*	0/15
$\mathbf{f_{12}}$	*2.5e+8*:1.3	*6.3e+6*:2.7	*6.3e+5*:6.3	*4.0e+1*:21	*1.6e-3*:101	15/15
	0.95(0.4)	1(0.9)	1.8(2)	∞	∞*20*	0/15
$\mathbf{f_{13}}$	*4.0e+2*:1.6	*2.5e+2*:3.1	*6.3e+1*:8.7	*1.0e+1*:23	*4.0e-6*:100	15/15
	2.4(3)	1.4(2)	2.4(3)	∞	∞*20*	0/15
$\mathbf{f_{14}}$	*1.6e+1*:1.4	*2.5e+0*:4.2	*1.0e+0*:7.4	*2.5e-2*:21	*1.0e-8*:101	15/15
	1.0(0.7)	1.8(2)	4.6(5)	∞	∞*20*	0/15
$\mathbf{f_{15}}$	*1.6e+2*:1.2	*4.0e+1*:4.7	*2.5e+1*:10	*1.0e+1*:37	*2.5e+0*:118	5/5
	2.3(2)	1.5(2)	1.7(2)	2.5(2)	∞*20*	0/15
$\mathbf{f_{16}}$	*1.0e+2*:1.1	*2.5e+1*:3.9	*1.6e+1*:6.5	*4.0e+0*:31	*2.5e-1*:127	5/5
	1.4(0.4)	1.6(2)	1.5(2)	2.1(2)	∞*20*	0/15
$\mathbf{f_{17}}$	*4.0e+1*:1.2	*1.0e+1*:2.7	*4.0e+0*:10	*2.5e+0*:28	*1.6e-1*:119	5/5
	1.3(0.4)	1.9(2)	3.4(3)	3.3(4)	∞*20*	0/15
$\mathbf{f_{18}}$	*4.0e+2*:1.2	*1.0e+2*:3.2	*4.0e+1*:7.2	*6.3e+0*:32	*1.6e+0*:104	5/5
	2.1(1)	1.5(1)	1.2(1)	1.4(2)	∞*20*	0/15
$\mathbf{f_{19}}$	*1.6e-1*:23	*1.0e-1*:26	*6.3e-2*:38	*4.0e-2*:40	*1.0e-2*:216	15/15
	13(13)	11(12)	7.8(8)	∞	∞*20*	0/15
$\mathbf{f_{20}}$	*4.0e+3*:1.9	*2.5e+2*:2.8	*4.0e+0*:6.3	*2.5e+0*:21	*6.3e-1*:139	15/15
	1.1(1)	3.4(4)	7.9(9)	14(16)	∞*20*	0/15
$\mathbf{f_{21}}$	*1.0e+1*:1.7	*6.3e+0*:2.6	*2.5e+0*:7.9	*1.6e+0*:30	*4.0e-1*:105	15/15
	1.9(2)	2.2(2)	1.3(1)	2.9(3)	∞*20*	0/15
$\mathbf{f_{22}}$	*4.0e+1*:1.3	*1.6e+1*:3.2	*6.3e+0*:9.3	*1.6e+0*:25	*1.0e-1*:168	15/15
	2.2(2)	1.7(2)	1.5(1)	1.4(2)	1.7(2)	1/15
$\mathbf{f_{23}}$	*4.0e+1*:1.5	*2.5e+1*:2.6	*1.0e+1*:7.8	*4.0e+0*:55	*2.5e+0*:103	5/5
	1.6(1)	1.1(1.0)	1.7(2)	0.74(0.8)	0.85(0.9)	3/15
$\mathbf{f_{24}}$	*4.0e+1*:1.1	*2.5e+1*:2.7	*1.6e+1*:7.7	*6.3e+0*:44	*2.5e+0*:275	5/5
	1.6(0.9)	1.6(1)	1.4(2)	3.2(4)	∞*20*	0/15

5-D

#FEs/D	0.5	1.2	3	10	50	#succ
$\mathbf{f_1}$	*2.5e+1*:4.8	*1.6e+1*:7.6	*1.0e-8*:12	*1.0e-8*:12	*1.0e-8*:12	15/15
	1.5(1)	1.9(2)	∞	∞	∞*50*	0/15
$\mathbf{f_2}$	*1.6e+6*:2.9	*4.0e+5*:11	*4.0e+4*:15	*6.3e+2*:58	*1.0e-8*:95	15/15
	0.95(0.7)	1.0(1)	11(12)	∞	∞*50*	0/15
$\mathbf{f_3}$	*1.6e+2*:4.1	*1.0e+2*:15	*6.3e+1*:23	*2.5e+1*:73	*1.0e+1*:716	15/15
	2.1(2)	1.8(1)	5.2(6)	∞	∞*50*	0/15
$\mathbf{f_4}$	*2.5e+2*:2.6	*1.6e+2*:10	*1.0e+2*:19	*4.0e+1*:65	*1.6e+1*:434	15/15
	2.4(2)	3.3(4)	6.5(7)	∞	∞*50*	0/15
$\mathbf{f_5}$	*6.3e+1*:4.0	*4.0e+1*:10	*1.0e-8*:10	*1.0e-8*:10	*1.0e-8*:10	15/15
	1.9(2)	6.5(6)	∞	∞	∞*50*	0/15
$\mathbf{f_6}$	*1.0e+5*:3.0	*2.5e+4*:8.4	*1.0e+2*:16	*2.5e+1*:54	*2.5e-1*:254	15/15
	1.6(2)	1.8(3)	8.3(8)	∞	∞*50*	0/15
$\mathbf{f_7}$	*1.6e+2*:4.2	*1.0e+2*:6.2	*2.5e+1*:20	*4.0e+0*:54	*1.0e+0*:324	15/15
	1.9(2)	2.0(2)	11(11)	14(14)	2.3(2)	1/15
$\mathbf{f_8}$	*1.0e+4*:4.6	*6.3e+3*:6.8	*1.0e+3*:18	*6.3e+1*:54	*1.6e+0*:258	15/15
	1.1(1)	2.0(3)	40(40)	∞	∞*50*	0/15
$\mathbf{f_9}$	*2.5e+1*:20	*1.6e+1*:26	*1.0e+1*:35	*4.0e+0*:62	*1.6e-2*:256	15/15
	∞	∞	∞	∞	∞*50*	0/15
$\mathbf{f_{10}}$	*2.5e+6*:2.9	*6.3e+5*:7.0	*2.5e+5*:17	*6.3e+3*:54	*2.5e+1*:297	15/15
	1.1(0.7)	1.3(1)	1.7(1)	∞	∞*50*	0/15
$\mathbf{f_{11}}$	*1.0e+6*:3.0	*6.3e+4*:6.2	*6.3e+2*:16	*6.3e+1*:74	*6.3e-1*:298	15/15
	1.8(2)	2.3(3)	14(17)	10(11)	∞*50*	0/15
$\mathbf{f_{12}}$	*4.0e+7*:3.6	*1.6e+7*:7.6	*4.0e+6*:19	*1.6e+4*:52	*1.0e+0*:268	15/15
	1.2(1)	2.9(3)	18(21)	∞	∞*50*	0/15
$\mathbf{f_{13}}$	*1.0e+3*:2.8	*6.3e+2*:8.4	*4.0e+2*:17	*6.3e+1*:52	*6.3e-2*:264	15/15
	2.5(3)	4.2(5)	9.4(10)	∞	∞*50*	0/15
$\mathbf{f_{14}}$	*1.6e+1*:3.0	*1.0e+1*:10	*6.3e+0*:15	*2.5e-1*:53	*1.0e-5*:251	15/15
	2.4(2)	1.4(1)	2.9(4)	∞	∞*50*	0/15
$\mathbf{f_{15}}$	*1.6e+2*:3.0	*1.0e+2*:13	*6.3e+1*:24	*4.0e+1*:55	*1.6e+1*:289	5/5
	2.8(2)	2.4(3)	8.7(10)	6.2(7)	∞*50*	0/15
$\mathbf{f_{16}}$	*4.0e+1*:4.8	*2.5e+1*:16	*1.6e+1*:46	*1.0e+1*:120	*4.0e+0*:334	15/15
	1.4(1)	1.6(2)	2.2(2)	∞	∞*50*	0/15
$\mathbf{f_{17}}$	*1.0e+1*:5.2	*6.3e+0*:26	*4.0e+0*:57	*2.5e+0*:110	*6.3e-1*:412	15/15
	4.0(4)	6.0(8)	5.9(7)	∞	∞*50*	0/15
$\mathbf{f_{18}}$	*6.3e+1*:3.4	*4.0e+1*:7.2	*2.5e+1*:20	*1.6e+1*:58	*1.6e+0*:318	15/15
	1.8(1)	3.5(4)	8.1(9)	∞	∞*50*	0/15
$\mathbf{f_{19}}$	*1.6e-1*:172	*1.0e-1*:242	*6.3e-2*:675	*4.0e-2*:3078	*2.5e-2*:4946	15/15
	∞	∞	∞	∞	∞*50*	0/15
$\mathbf{f_{20}}$	*6.3e+3*:5.1	*4.0e+3*:8.4	*4.0e+1*:15	*2.5e+0*:69	*1.0e+0*:851	15/15
	2.0(2)	2.6(3)	47(51)	∞	∞*50*	0/15
$\mathbf{f_{21}}$	*4.0e+1*:3.9	*2.5e+1*:11	*1.6e+1*:31	*6.3e+0*:73	*1.6e+0*:347	5/5
	1.2(0.9)	1.5(2)	1.2(1)	4.9(5)	∞*50*	0/15
$\mathbf{f_{22}}$	*6.3e+1*:3.6	*4.0e+1*:15	*2.5e+1*:32	*1.0e+1*:71	*1.6e+0*:341	5/5
	2.3(3)	2.0(2)	2.5(3)	11(11)	∞*50*	0/15
$\mathbf{f_{23}}$	*1.0e+1*:3.0	*6.3e+0*:9.0	*4.0e+0*:33	*2.5e+0*:84	*1.0e+0*:518	15/15
	1.6(2)	1.6(2)	5.1(5)	∞	∞*50*	0/15
$\mathbf{f_{24}}$	*6.3e+1*:15	*4.0e+1*:37	*4.0e+1*:37	*2.5e+1*:118	*1.6e+1*:692	15/15
	3.3(3)	∞	∞	∞	∞*50*	0/15

Table 1: 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.

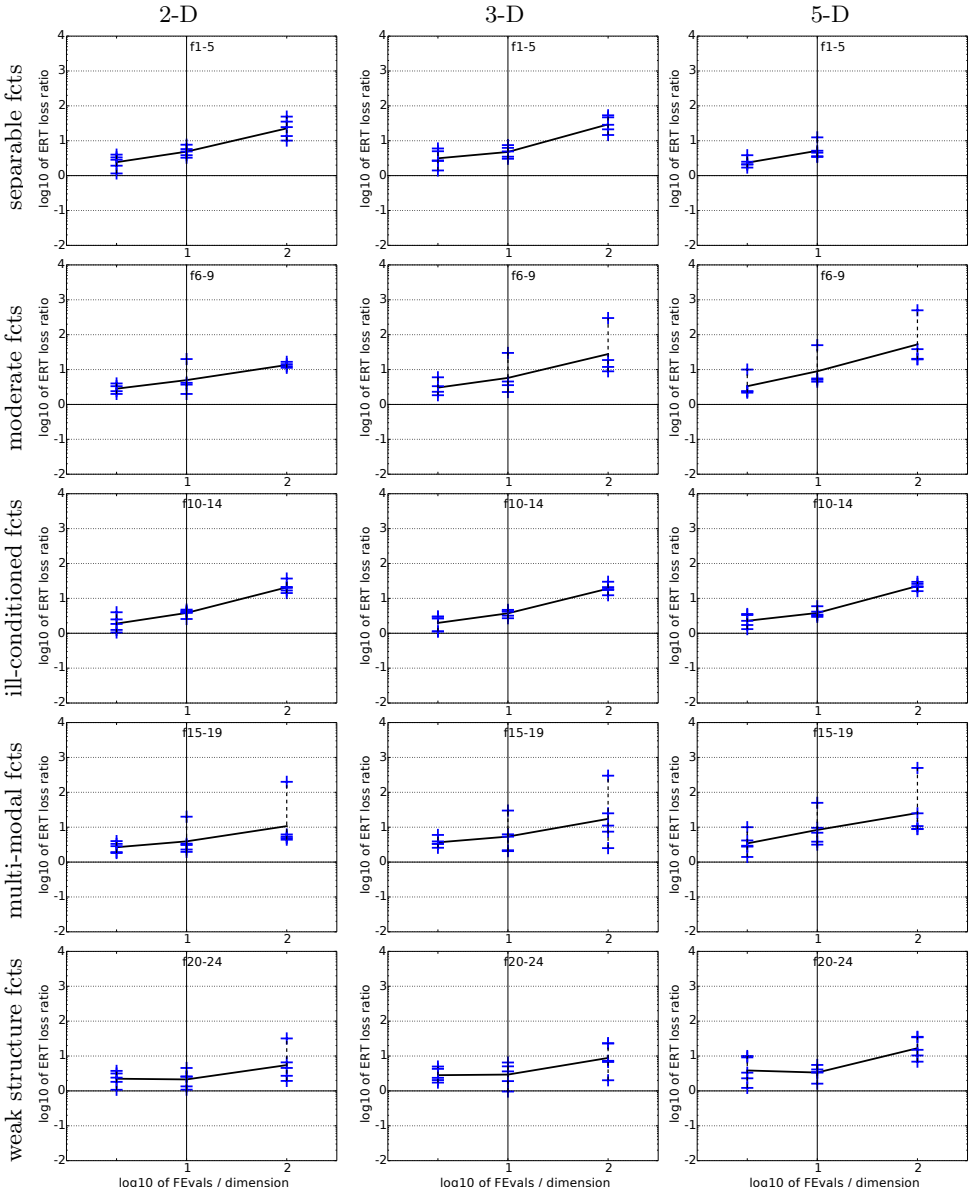


Figure 4: ERT **loss ratios (see Figure 3 for details). Each cross (+) represents a single function, the line is the geometric mean.**