



HAL
open science

Benchmarking a BI-Population CMA-ES on the BBOB-2009 Noisy Testbed

Nikolaus Hansen

► **To cite this version:**

Nikolaus Hansen. Benchmarking a BI-Population CMA-ES on the BBOB-2009 Noisy Testbed. ACM-GECCO Genetic and Evolutionary Computation Conference, Jul 2009, Montreal, Canada. inria-00382101

HAL Id: inria-00382101

<https://inria.hal.science/inria-00382101>

Submitted on 7 May 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Benchmarking a BI-Population CMA-ES on the BBOB-2009 Noisy Testbed

Nikolaus Hansen
Microsoft Research–INRIA Joint Centre
28 rue Jean Rostand
91893 Orsay Cedex, France
Nikolaus.Hansen@inria.fr

ABSTRACT

We benchmark the BI-population CMA-ES on the BBOB-2009 noisy functions testbed. BI-population refers to a multistart strategy with equal budgets for two interlaced restart strategies, one with an increasing population size and one with varying small population sizes. The latter is presumably of little use on a noisy testbed. The BI-population CMA-ES could solve 29, 27 and 26 out of 30 functions in search space dimension 5, 10 and 20 respectively. The time to find the solution ranges between $100D$ and $10^5 D^2$ objective function evaluations, where D is the search space dimension.

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, Direct search, Evolutionary computation, CMA-ES

1. INTRODUCTION

The *covariance matrix adaptation evolution strategy* (CMA-ES) [2, 6, 7] is a stochastic, population-based search method in continuous search spaces, aiming at minimizing an objective function $f : \mathbb{R}^D \rightarrow \mathbb{R}$ in a black-box scenario. In this paper, the $(\mu/\mu_w, \lambda)$ -CMA-ES is applied in a multistart strategy and benchmarked on 30 noisy functions. The multistart consists of two interlacing strategies, one with increasing population size, the other with varying small population size. The algorithm is given in a complementing paper in the same proceedings [3].

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 '09, July 8–12, 2009, Montréal Québec, Canada.
Copyright 2009 ACM 978-1-60558-505-5/09/07 ...\$5.00.

2. ALGORITHM AND PARAMETER SETTINGS

The algorithm and all parameters are described in [3]. The following parameters have been chosen differently, where λ denotes the offspring population size.

$\text{MaxIter} = 1000 + 500(D + 3)^2/\sqrt{\lambda}$ has been chosen ten times larger, as the noisy functions are expected to be more difficult to solve.

$c_1 = \frac{0.4}{(D+1.3)^2 + \mu_w}$ and $c_\mu = \min\left(1 - c_1, 0.4 \frac{\mu_w - 2 + 1/\mu_w}{(D+2)^2 + \mu_w}\right)$, the learning rates for the covariance matrix have been chosen five times smaller than by default. With default learning rates, the learning of the covariance matrix is the algorithm component that is most susceptible to uncertainties in the selection. Surprisingly, this change did not produce a striking improvement, but slightly more consistent results (e.g. f_{105} , f_{108} and f_{117} are solved also in 40-D).

Restarts are launched until $10^6 D$ function evaluations are exceeded. The stagnation termination criterion

Stagnation: terminate a run, if the median of the 20 newest values is not smaller than the median of the 20 oldest values, respectively, in the two arrays containing the best function values and the median function values of the last $\lceil 0.2t + 120 + 30D/\lambda \rceil$ iterations, where t denotes the iteration counter,

turns out to be crucial for the noisy testbed. Most other standard termination criteria regularly fail. We presume that restarts with small population size are less valuable on a noisy testbed, which leaves yet room for improvement.

The same parameter setting is used for all functions and therefore the crafting effort according to [4] is $\text{CrE} = 0$.

3. SUCCESSFUL POPULATION SIZE

We investigate the population sizes of the final successful runs. In Table 1 minimal, median (the larger in case of even data) and maximal population size are given for 10-D and 20-D. Functions solved with default offspring population size, $\lambda = 10$ and 12, are the sphere function with moderate noise and about half of the functions with Cauchy noise (function numbers in *italics*). Otherwise is the typical population size $10D$ or larger.

Table 2 tabulates minimal, median (the larger in case of even data) and maximal initial step-size σ^0 of the final successful runs, whenever $\sigma^0 < 2$ in at least one successful case.

Table 1: Final, successful population sizes. If less than three successful trials included restarts, only the median (default population size) is given

f	$D = 10$			$D = 20$		
	min	med	max	min	med	max
101		10			12	
102		10			12	
103		10			12	
104	20	20	40	96	96	192
105	40	80	80	192	192	384
106		10			12	
107	20	20	20	24	24	48
108	80	160	160	192	192	384
109		10			12	
110	-	-	-	-	-	-
111	-	-	-	-	-	-
112		10			12	
113	40	80	320	192	384	384
114	160	320	640	286	768	768
115	40	80	160	185	192	384
116	80	160	160	192	384	384
117	160	320	640	768	768	768
118		10			12	
119	160	160	160	384	384	384
120	320	640	1280	1536	1536	1536
121		10			12	
122	80	160	320	384	768	768
123	640	1280	1280	-	-	-
124	40	80	160	96	192	192
125	640	1280	2560	3072	6144	6144
126	-	-	-	-	-	-
127	239	640	1280	384	1536	3072
128	11	40	320	37	192	768
129	80	118	1280	1301	3072	6144
130	10	26	166	12	24	192

Table 2: Initial step-size σ^0 of successful restarts for functions, where $\sigma^0 < 2$ was successful at least once

f	$D = 10$			f	$D = 20$		
	min	med	max		min	med	max
115	0.126	2.0	2.0	114	1.06	2.0	2.0
125	0.72	2.0	2.0	115	0.068	2.0	2.0
127	0.158	2.0	2.0	122	0.2	2.0	2.0
128	0.28	1.8	2.0	128	0.64	1.56	2.0
129	0.8	2.0	2.0	129	1.42	2.0	2.0
130	0.06	0.64	2.0	130	0.042	0.38	2.0

The data neither rule out nor suggest that f_{130} might benefit from a small initial step-size. Overall, as expected, the small initial step-size is of little use and, if anything, rather disadvantageous on the BBOB-2009 noisy testbed.

4. RESULTS AND DISCUSSION

The results of CPU timing experiments are given in [3]: using Matlab, about $2-3 \times 10^{-4}$ seconds per function evaluation are needed on the BBOB-2009 f_8 function for up to 40-D. Results from the performance experiments according to [4] on the benchmarks functions given in [1, 5] are presented in Figures 1 and 2 and in Tables 3 and 4.

The number of solved functions are 30, 30, 29, 27, 26 and

24 out of 30 functions in 2, 3, 5, 10, 20 and 40-D, respectively.

The typical scaling of the running time (number of function evaluations) is quadratic with the dimension (see Figure 1). With the moderate noise models and with the Cauchy noise model the sphere function can be solved in linear time. More severe noise impairs the scaling behavior by one order of magnitude to quadratic. Similarly, on the Rosenbrock function the severe noise impairs the scaling by one order from roughly quadratic to roughly cubic. Here, the observed failure is presumably due to a too small maximum number of function evaluations allowed.

The expected running times appear rather uniform on the log-scale between $100D$ and $10^5 D^2$ function evaluations (upper figures in Figure 2). The graphs suggest that, in higher dimension, with more function evaluations even more functions can be solved.

Acknowledgments

The author would like to acknowledge the great and hard work of the BBOB team with particular kudos to Raymond Ros, Steffen Finck and Anne Auger, and Anne Auger and Marc Schoenauer for their kind and persistent support.

5. REFERENCES

- [1] S. Finck, N. Hansen, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Presentation of the noisy functions. Technical Report 2009/21, Research Center PPE, 2009.
- [2] N. Hansen. The CMA evolution strategy: a comparing review. In J. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, editors, *Towards a new evolutionary computation. Advances on estimation of distribution algorithms*, pages 75–102. Springer, 2006.
- [3] N. Hansen. Benchmarking a BI-population CMA-ES on the BBOB-2009 function testbed. In *Workshop Proceedings of the GECCO Genetic and Evolutionary Computation Conference*. ACM, 2009.
- [4] N. Hansen, A. Auger, S. Finck, and R. Ros. Real-parameter black-box optimization benchmarking 2009: Experimental setup. Technical Report RR-6828, INRIA, 2009.
- [5] N. Hansen, S. Finck, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Noisy functions definitions. Technical Report RR-6869, INRIA, 2009.
- [6] N. Hansen and S. Kern. Evaluating the CMA evolution strategy on multimodal test functions. In X. Yao et al., editors, *Parallel Problem Solving from Nature - PPSN VIII, LNCS 3242*, pages 282–291. Springer, 2004.
- [7] N. Hansen, A. Niederberger, L. Guzzella, and P. Koumoutsakos. A method for handling uncertainty in evolutionary optimization with an application to feedback control of combustion. *IEEE Transactions on Evolutionary Computation*, 13(1):180–197, 2009.

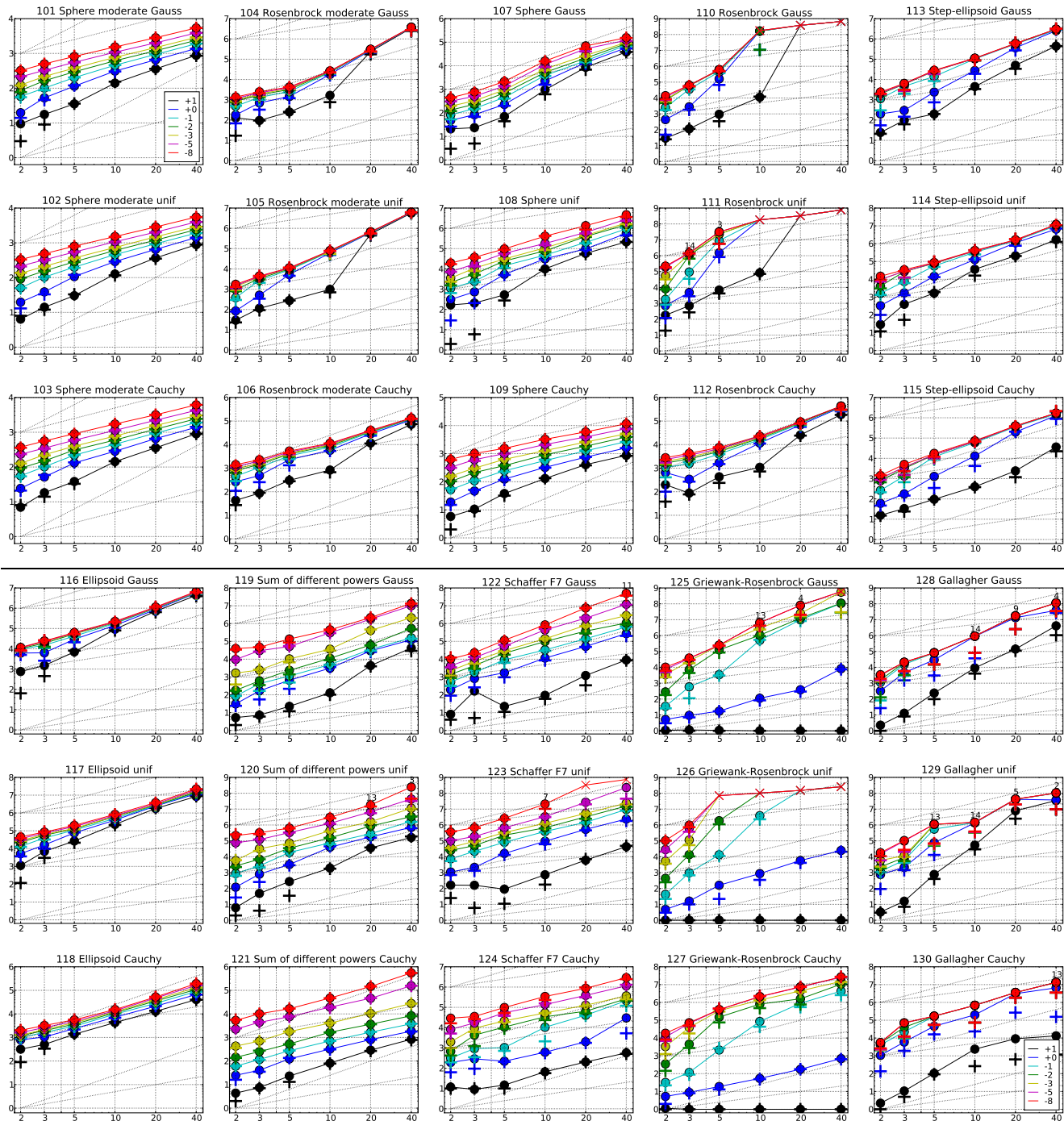


Figure 1: Expected Running Time (ERT, ●) to reach $f_{\text{opt}} + \Delta f$ and median number of function evaluations of successful trials (+), shown for $\Delta f = 10, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-5}, 10^{-8}$ (the exponent is given in the legend of f_{101} and f_{130}) versus dimension in log-log presentation. The $\text{ERT}(\Delta f)$ equals to $\#FEs(\Delta f)$ divided by the number of successful trials, where a trial is successful if $f_{\text{opt}} + \Delta f$ was surpassed during the trial. The $\#FEs(\Delta f)$ are the total number of function evaluations while $f_{\text{opt}} + \Delta f$ was not surpassed during the trial from all respective trials (successful and unsuccessful), and f_{opt} denotes the optimal function value. Crosses (×) indicate the total number of function evaluations $\#FEs(-\infty)$. Numbers above ERT-symbols indicate the number of successful trials. Annotated numbers on the ordinate are decimal logarithms. Additional grid lines show linear and quadratic scaling.

f_{101} in 5-D, N=15, mFE=930					f_{101} in 20-D, N=15, mFE=3014					f_{102} in 5-D, N=15, mFE=882					f_{102} in 20-D, N=15, mFE=3122						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	3.5e1	2.9e1	4.2e1	3.5e1	15	3.6e2	3.4e2	3.8e2	3.6e2	10	15	3.0e1	2.5e1	3.6e1	3.0e1	15	3.7e2	3.5e2	3.9e2	3.7e2
1	15	1.1e2	1.0e2	1.2e2	1.1e2	15	6.5e2	6.3e2	6.7e2	6.5e2	1	15	1.1e2	9.6e1	1.2e2	1.1e2	15	6.5e2	6.3e2	6.6e2	6.5e2
1e-1	15	2.0e2	1.9e2	2.1e2	2.0e2	15	9.4e2	9.2e2	9.5e2	9.4e2	1e-1	15	2.0e2	1.9e2	2.1e2	2.0e2	15	9.4e2	9.2e2	9.6e2	9.4e2
1e-3	15	3.8e2	3.6e2	3.9e2	3.8e2	15	1.5e3	1.4e3	1.5e3	1.5e3	1e-3	15	3.7e2	3.6e2	3.8e2	3.7e2	15	1.5e3	1.5e3	1.5e3	1.5e3
1e-5	15	5.6e2	5.4e2	5.7e2	5.6e2	15	2.0e3	2.0e3	2.0e3	2.0e3	1e-5	15	5.4e2	5.3e2	5.6e2	5.4e2	15	2.0e3	2.0e3	2.1e3	2.0e3
1e-8	15	8.2e2	8.0e2	8.3e2	8.2e2	15	2.8e3	2.8e3	2.9e3	2.8e3	1e-8	15	8.0e2	7.8e2	8.2e2	8.0e2	15	2.9e3	2.8e3	2.9e3	2.9e3
f_{103} in 5-D, N=15, mFE=994					f_{103} in 20-D, N=15, mFE=3386					f_{104} in 5-D, N=15, mFE=7644					f_{104} in 20-D, N=15, mFE=668758						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	3.8e1	3.0e1	4.7e1	3.8e1	15	3.6e2	3.4e2	3.7e2	3.6e2	10	15	2.5e2	2.4e2	2.7e2	2.5e2	15	2.4e5	2.0e5	2.8e5	2.4e5
1	15	1.3e2	1.2e2	1.4e2	1.3e2	15	6.6e2	6.4e2	6.7e2	6.6e2	1	15	1.4e3	1.2e3	1.7e3	1.4e3	15	2.8e5	2.3e5	3.3e5	2.8e5
1e-1	15	2.2e2	2.1e2	2.3e2	2.2e2	15	9.5e2	9.3e2	9.6e2	9.5e2	1e-1	15	2.5e3	2.3e3	2.8e3	2.5e3	15	2.9e5	2.4e5	3.4e5	2.9e5
1e-3	15	4.0e2	3.9e2	4.1e2	4.0e2	15	1.6e3	1.5e3	1.6e3	1.6e3	1e-3	15	3.5e3	3.3e3	3.8e3	3.5e3	15	3.0e5	2.5e5	3.5e5	3.0e5
1e-5	15	5.9e2	5.8e2	6.1e2	5.9e2	15	2.2e3	2.2e3	2.2e3	2.2e3	1e-5	15	3.9e3	3.6e3	4.2e3	3.9e3	15	3.0e5	2.5e5	3.5e5	3.0e5
1e-8	15	9.1e2	8.9e2	9.2e2	9.1e2	15	3.2e3	3.1e3	3.2e3	3.2e3	1e-8	15	4.3e3	4.0e3	4.6e3	4.3e3	15	3.1e5	2.6e5	3.6e5	3.1e5
f_{105} in 5-D, N=15, mFE=19372					f_{105} in 20-D, N=15, mFE=1.28e6					f_{106} in 5-D, N=15, mFE=11108					f_{106} in 20-D, N=15, mFE=86380						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	2.8e2	2.6e2	2.9e2	2.8e2	15	5.3e5	4.4e5	6.1e5	5.3e5	10	15	3.0e2	2.8e2	3.3e2	3.0e2	15	1.2e4	1.1e4	1.3e4	1.2e4
1	15	5.2e3	4.4e3	6.2e3	5.2e3	15	6.1e5	5.2e5	7.1e5	6.1e5	1	15	2.3e3	1.6e3	3.0e3	2.3e3	15	2.9e4	2.3e4	3.5e4	2.9e4
1e-1	15	8.9e3	7.8e3	1.0e4	8.9e3	15	6.3e5	5.3e5	7.3e5	6.3e5	1e-1	15	3.3e3	2.6e3	4.1e3	3.3e3	15	3.3e4	2.7e4	3.9e4	3.3e4
1e-3	15	1.0e4	9.2e3	1.2e4	1.0e4	15	6.5e5	5.5e5	7.5e5	6.5e5	1e-3	15	4.4e3	3.6e3	5.2e3	4.4e3	15	3.7e4	3.1e4	4.3e4	3.7e4
1e-5	15	1.1e4	9.7e3	1.2e4	1.1e4	15	6.6e5	5.7e5	7.6e5	6.6e5	1e-5	15	4.8e3	4.0e3	5.6e3	4.8e3	15	3.9e4	3.3e4	4.5e4	3.9e4
1e-8	15	1.1e4	1.0e4	1.3e4	1.1e4	15	6.8e5	5.8e5	7.8e5	6.8e5	1e-8	15	5.3e3	4.5e3	6.2e3	5.3e3	15	4.0e4	3.5e4	4.6e4	4.0e4
f_{107} in 5-D, N=15, mFE=3162					f_{107} in 20-D, N=15, mFE=141332					f_{108} in 5-D, N=15, mFE=200594					f_{108} in 20-D, N=15, mFE=3.02e6						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	7.0e1	4.6e1	9.6e1	7.0e1	15	8.6e3	6.9e3	1.0e4	8.6e3	10	15	5.3e2	3.1e2	7.7e2	5.3e2	15	5.8e4	5.1e4	6.5e4	5.8e4
1	15	2.3e2	1.9e2	2.7e2	2.3e2	15	1.4e4	1.1e4	1.6e4	1.4e4	1	15	5.4e3	4.2e3	6.5e3	5.4e3	15	9.7e4	8.3e4	1.1e5	9.7e4
1e-1	15	4.5e2	4.0e2	5.1e2	4.5e2	15	1.6e4	1.4e4	1.9e4	1.6e4	1e-1	15	1.4e4	1.2e4	1.8e4	1.4e4	15	2.0e5	1.7e5	2.3e5	2.0e5
1e-3	15	9.4e2	8.8e2	1.0e3	9.4e2	15	2.7e4	2.4e4	3.0e4	2.7e4	1e-3	15	3.1e4	2.6e4	3.6e4	3.1e4	15	4.5e5	3.9e5	5.0e5	4.5e5
1e-5	15	1.4e3	1.3e3	1.4e3	1.4e3	15	5.2e4	4.2e4	6.4e4	5.2e4	1e-5	15	5.9e4	5.3e4	6.4e4	5.9e4	15	6.3e5	5.6e5	7.0e5	6.3e5
1e-8	15	2.2e3	2.0e3	2.3e3	2.2e3	15	6.9e4	5.7e4	8.2e4	6.9e4	1e-8	15	9.7e4	8.3e4	1.1e5	9.7e4	15	1.4e6	1.1e6	1.6e6	1.4e6
f_{109} in 5-D, N=15, mFE=1922					f_{109} in 20-D, N=15, mFE=6878					f_{110} in 5-D, N=15, mFE=1.33e6					f_{110} in 20-D, N=15, mFE=3.09e7						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	3.8e1	3.2e1	4.4e1	3.8e1	15	4.1e2	3.8e2	4.3e2	4.1e2	10	15	9.5e2	4.1e2	1.5e3	9.5e2	0	<i>17e+0</i>	<i>16e+0</i>	<i>17e+0</i>	<i>1.8e7</i>
1	15	1.2e2	1.1e2	1.4e2	1.2e2	15	7.5e2	7.2e2	7.8e2	7.5e2	1	15	1.6e5	9.9e4	2.2e5	1.6e5					
1e-1	15	2.4e2	2.3e2	2.7e2	2.4e2	15	1.3e3	1.2e3	1.3e3	1.3e3	1e-1	15	4.5e5	3.3e5	5.7e5	4.5e5					
1e-3	15	6.2e2	5.9e2	6.6e2	6.2e2	15	2.6e3	2.5e3	2.7e3	2.6e3	1e-3	15	5.9e5	4.9e5	7.0e5	5.9e5					
1e-5	15	1.0e3	9.5e2	1.1e3	1.0e3	15	3.9e3	3.8e3	4.1e3	3.9e3	1e-5	15	6.0e5	5.0e5	7.1e5	6.0e5					
1e-8	15	1.5e3	1.5e3	1.6e3	1.5e3	15	6.0e3	5.8e3	6.1e3	6.0e3	1e-8	15	6.2e5	5.1e5	7.3e5	6.2e5					
f_{111} in 5-D, N=15, mFE=1.11e7					f_{111} in 20-D, N=15, mFE=2.51e7					f_{112} in 5-D, N=15, mFE=9378					f_{112} in 20-D, N=15, mFE=187012						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	6.9e3	4.4e3	9.5e3	6.9e3	0	<i>18e+0</i>	<i>17e+0</i>	<i>18e+0</i>	<i>1.8e7</i>	10	15	4.3e2	2.6e2	6.1e2	4.3e2	15	2.6e4	2.4e4	2.7e4	2.6e4
1	14	1.5e6	9.3e5	2.2e6	1.1e6						1	15	1.7e3	1.4e3	2.0e3	1.7e3	15	6.9e4	5.6e4	8.1e4	6.9e4
1e-1	8	8.8e6	6.3e6	1.3e7	4.8e6						1e-1	15	4.0e3	3.6e3	4.3e3	4.0e3	15	7.9e4	6.7e4	9.1e4	7.9e4
1e-3	4	2.3e7	1.4e7	5.1e7	5.1e6						1e-3	15	5.8e3	5.4e3	6.1e3	5.8e3	15	8.6e4	7.4e4	1.0e5	8.6e4
1e-5	3	3.1e7	1.8e7	8.8e7	6.8e6						1e-5	15	6.6e3	6.3e3	6.9e3	6.6e3	15	9.0e4	7.7e4	1.0e5	9.0e4
1e-8	3	3.1e7	1.9e7	9.3e7	7.0e6						1e-8	15	7.6e3	7.2e3	7.9e3	7.6e3	15	9.3e4	8.0e4	1.1e5	9.3e4
f_{113} in 5-D, N=15, mFE=86966					f_{113} in 20-D, N=15, mFE=795352					f_{114} in 5-D, N=15, mFE=140043					f_{114} in 20-D, N=15, mFE=2.53e6						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	2.0e2	1.7e2	2.3e2	2.0e2	15	5.0e4	3.7e4	6.3e4	5.0e4	10	15	1.7e3	1.3e3	2.1e3	1.7e3	15	2.1e5	1.8e5	2.4e5	2.1e5
1	15	2.4e3	1.4e3	3.5e3	2.4e3	15	3.6e5	3.1e5	4.2e5	3.6e5	1	15	1.5e4	1.2e4	1.8e4	1.5e4	15	1.1e6	9.5e5	1.3e6	1.1e6
1e-1	15	1.4e4	1.0e4	1.9e4	1.4e4	15	5.6e5	5.1e5	6.1e5	5.6e5	1e-1	15	5.6e4	4.7e4	6.7e4	5.6e4	15	1.4e6	1.3e6	1.6e6	1.4e6
1e-3	15	2.7e4	1.9e4	3.5e4	2.7e4	15	5.9e5	5.3e5	6.4e5	5.9e5	1e-3	15	8.3e4	6.9e4	9.7e4	8.3e4	15	1.6e6	1.4e6	1.8e6	1.6e6
1e-5	15	2.7e4	1.9e4	3.5e4	2.7e4	15	5.9e5	5.3e5	6.4e5	5.9e5	1e-5	15	8.3e4	6.9e4	9.7e4	8.3e4	15	1.6e6	1.4e6	1.8e6	1.6e6
1e-8	15	2.8e4	2.1e4	3.6e4	2.8e4	15	6.0e5	5.4e5	6.5e5	6.0e5	1e-8	15	8.7e4	7.2e4	1.0e5	8.7e4	15	1.7e6	1.4e6	1.9e6	1.7e6
f_{115} in 5-D, N=15, mFE=45496					f_{115} in 20-D, N=15, mFE=678416					f_{116} in 5-D, N=15, mFE=115497					f_{116} in 20-D, N=15, mFE=1.69e6						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	9.6e1	7.9e1	1.1e2	9.6e1</																

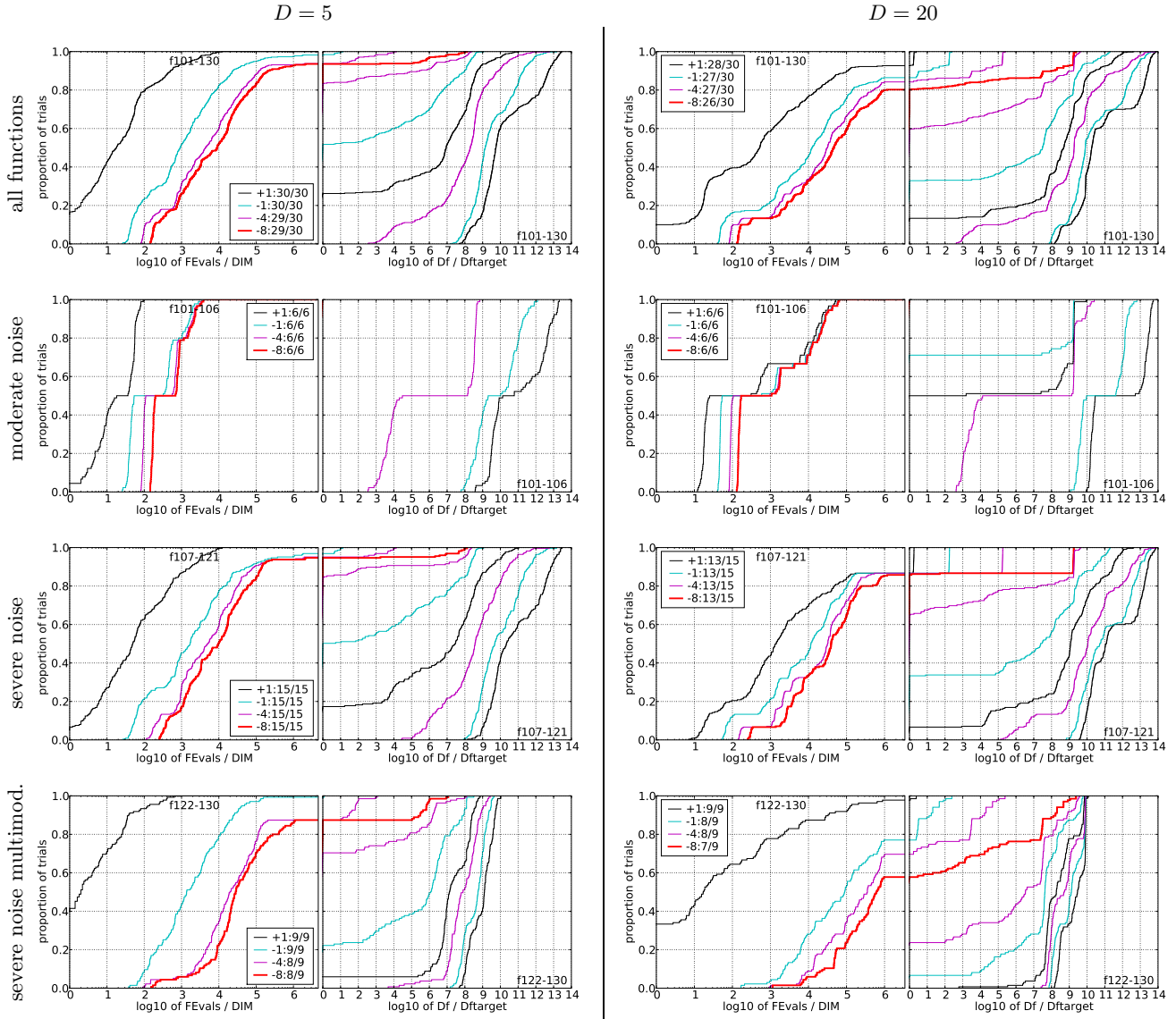


Figure 2: Empirical cumulative distribution functions (ECDFs), plotting the fraction of trials versus running time (left subplots) or versus Δf (right subplots). The thick red line represents the best achieved results. Left subplots: ECDF of the running time (number of function evaluations), divided by search space dimension D , to fall below $f_{\text{opt}} + \Delta f$ with $\Delta f = 10^k$, where k is the first value in the legend. Right subplots: ECDF of the best achieved Δf divided by 10^k (upper left lines in continuation of the left subplot), and best achieved Δf divided by 10^{-8} for running times of $D, 10D, 100D \dots$ function evaluations (from right to left cycling black-cyan-magenta). Top row: all results from all functions; second row: moderate noise functions; third row: severe noise functions; fourth row: severe noise and highly-multimodal functions. The legends indicate the number of functions that were solved in at least one trial. FEvals denotes number of function evaluations, D and DIM denote search space dimension, and Δf and Df denote the difference to the optimal function value.

f_{121} in 5-D, N=15, mFE=17658					f_{121} in 20-D, N=15, mFE=164174					f_{122} in 5-D, N=15, mFE=226165					f_{122} in 20-D, N=15, mFE=1.55e7						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	2.3e1	1.6e1	3.0e1	2.3e1	15	2.9e2	2.6e2	3.2e2	2.9e2	15	2.2e1	1.2e1	3.3e1	2.2e1	15	1.2e3	4.7e2	2.0e3	1.2e3	
1	15	1.2e2	1.1e2	1.4e2	1.2e2	15	8.1e2	7.7e2	8.4e2	8.1e2	1	15	1.7e3	1.3e3	2.2e3	1.7e3	15	5.2e4	4.5e4	5.9e4	5.2e4
1e-1	15	2.7e2	2.5e2	2.9e2	2.7e2	15	1.7e3	1.6e3	1.8e3	1.7e3	1e-1	15	9.2e3	7.1e3	1.1e4	9.2e3	15	1.4e5	1.2e5	1.6e5	1.4e5
1e-3	15	1.8e3	1.6e3	2.0e3	1.8e3	15	1.1e4	1.0e4	1.1e4	1.1e4	1e-3	15	3.0e4	2.6e4	3.4e4	3.0e4	15	7.9e5	6.5e5	9.5e5	7.9e5
1e-5	15	7.6e3	7.3e3	7.9e3	7.6e3	15	4.6e4	4.5e4	4.7e4	4.6e4	1e-5	15	5.4e4	4.6e4	6.1e4	5.4e4	15	2.0e6	1.8e6	2.2e6	2.0e6
1e-8	15	1.7e4	1.6e4	1.7e4	1.7e4	15	1.5e5	1.4e5	1.5e5	1.5e5	1e-8	15	1.2e5	1.0e5	1.3e5	1.2e5	15	7.5e6	6.2e6	8.9e6	7.5e6
f_{123} in 5-D, N=15, mFE=5.43e6					f_{123} in 20-D, N=15, mFE=2.56e7					f_{124} in 5-D, N=15, mFE=403943					f_{124} in 20-D, N=15, mFE=1.12e6						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	9.1e1	3.5e1	1.6e2	9.1e1	15	6.1e3	4.8e3	7.3e3	6.1e3	10	15	1.5e1	1.1e1	1.9e1	1.5e1	15	2.0e2	1.8e2	2.3e2	2.0e2
1	15	1.6e4	1.3e4	1.9e4	1.6e4	15	5.3e5	4.2e5	6.4e5	5.3e5	1	15	2.2e2	2.0e2	2.4e2	2.2e2	15	2.0e3	1.7e3	2.3e3	2.0e3
1e-1	15	8.2e4	6.6e4	9.8e4	8.2e4	15	1.5e6	1.2e6	1.8e6	1.5e6	1e-1	15	1.0e3	7.3e2	1.4e3	1.0e3	15	4.1e4	3.2e4	5.0e4	4.1e4
1e-3	15	3.4e5	2.8e5	4.0e5	3.4e5	15	5.3e6	4.6e6	6.0e6	5.3e6	1e-3	15	2.2e4	1.7e4	2.6e4	2.2e4	15	1.3e5	1.1e5	1.5e5	1.3e5
1e-5	15	6.7e5	5.7e5	7.8e5	6.7e5	10	2.7e7	2.3e7	3.4e7	1.9e7	1e-5	15	5.4e4	4.5e4	6.5e4	5.4e4	15	3.9e5	3.1e5	4.7e5	3.9e5
1e-8	15	2.6e6	2.1e6	3.1e6	2.6e6	0	<i>36e-7</i>	<i>61e-9</i>	<i>16e-6</i>	2.0e7	1e-8	15	9.8e4	7.0e4	1.3e5	9.8e4	15	8.7e5	7.8e5	9.6e5	8.7e5
f_{125} in 5-D, N=15, mFE=555113					f_{125} in 20-D, N=15, mFE=3.35e7					f_{126} in 5-D, N=15, mFE=6.03e6					f_{126} in 20-D, N=15, mFE=1.24e7						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	1.1e0	1.0e0	1.1e0	1.1e0	15	1.0e0	1.0e0	1.0e0	1.0e0	10	15	1.0e0	1.0e0	1.0e0	15	1.0e0	1.0e0	1.0e0	1.0e0	
1	15	1.7e1	1.3e1	2.2e1	1.7e1	15	3.8e2	3.1e2	4.6e2	3.8e2	1	15	1.6e2	4.0e1	2.8e2	1.6e2	15	5.8e3	4.6e3	6.9e3	5.8e3
1e-1	15	3.4e3	2.6e3	4.4e3	3.4e3	15	9.8e6	8.2e6	1.1e7	9.8e6	1e-1	15	1.3e4	1.0e4	1.6e4	1.3e4	0	<i>30e-2</i>	<i>24e-2</i>	<i>32e-2</i>	5.6e6
1e-3	15	2.4e5	2.0e5	2.8e5	2.4e5	10	2.5e7	1.9e7	3.4e7	1.5e7	1e-3	0	<i>67e-4</i>	<i>13e-4</i>	<i>98e-4</i>	2.8e6	
1e-5	15	2.4e5	2.0e5	2.8e5	2.4e5	4	8.0e7	5.3e7	1.6e8	2.3e7	1e-5	
1e-8	15	2.5e5	2.1e5	2.9e5	2.5e5	4	8.1e7	5.2e7	1.7e8	2.3e7	1e-8	
f_{127} in 5-D, N=15, mFE=1.03e6					f_{127} in 20-D, N=15, mFE=1.60e7					f_{128} in 5-D, N=15, mFE=666537					f_{128} in 20-D, N=15, mFE=2.37e7						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	1.0e0	1.0e0	1.0e0	1.0e0	15	1.0e0	1.0e0	1.0e0	1.0e0	10	15	2.4e2	9.4e1	3.9e2	2.4e2	15	1.4e5	8.7e4	2.0e5	1.4e5
1	15	1.9e1	1.3e1	2.5e1	1.9e1	15	1.8e2	1.6e2	1.9e2	1.8e2	1	15	2.9e4	8.9e3	5.5e4	2.9e4	10	1.3e7	8.2e6	2.0e7	8.9e6
1e-1	15	2.1e3	1.7e3	2.5e3	2.1e3	15	9.0e5	5.9e5	1.2e6	9.0e5	1e-1	15	8.2e4	2.8e4	1.4e5	8.2e4	9	1.7e7	1.1e7	2.5e7	1.2e7
1e-3	15	3.4e5	2.5e5	4.3e5	3.4e5	15	4.4e6	3.7e6	5.2e6	4.4e6	1e-3	15	8.2e4	2.8e4	1.4e5	8.2e4	9	1.7e7	1.2e7	2.5e7	1.2e7
1e-5	15	3.9e5	3.0e5	4.8e5	3.9e5	15	7.3e6	6.1e6	8.6e6	7.3e6	1e-5	15	8.2e4	2.8e4	1.4e5	8.2e4	9	1.7e7	1.1e7	2.5e7	1.2e7
1e-8	15	4.0e5	3.1e5	4.9e5	4.0e5	15	7.5e6	6.2e6	8.8e6	7.5e6	1e-8	15	8.3e4	2.8e4	1.4e5	8.3e4	9	1.7e7	1.2e7	2.6e7	1.2e7
f_{129} in 5-D, N=15, mFE=6.82e6					f_{129} in 20-D, N=15, mFE=2.27e7					f_{130} in 5-D, N=15, mFE=808354					f_{130} in 20-D, N=15, mFE=1.62e7						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	7.7e2	4.8e2	1.1e3	7.7e2	12	7.8e6	5.3e6	1.1e7	7.4e6	10	15	1.1e2	8.8e1	1.2e2	1.1e2	15	9.1e3	4.2e3	1.4e4	9.1e3
1	15	7.6e4	2.6e4	1.4e5	7.6e4	5	4.1e7	2.6e7	8.0e7	1.3e7	1	15	4.6e4	2.4e4	7.0e4	4.6e4	15	3.1e6	1.5e6	5.1e6	3.1e6
1e-1	14	5.5e5	5.7e4	1.1e6	5.5e5	5	4.2e7	2.6e7	8.2e7	1.3e7	1e-1	15	1.7e5	9.7e4	2.4e5	1.7e5	15	3.6e6	1.8e6	5.3e6	3.6e6
1e-3	13	1.1e6	9.5e4	2.0e6	1.1e6	5	4.2e7	2.7e7	8.2e7	1.3e7	1e-3	15	1.7e5	9.4e4	2.4e5	1.7e5	15	3.6e6	1.9e6	5.3e6	3.6e6
1e-5	13	1.1e6	9.1e4	2.1e6	1.1e6	5	4.2e7	2.6e7	7.8e7	1.3e7	1e-5	15	1.7e5	9.4e4	2.5e5	1.7e5	15	3.6e6	2.0e6	5.4e6	3.6e6
1e-8	13	1.1e6	9.7e4	2.0e6	1.1e6	5	4.3e7	2.7e7	8.0e7	1.4e7	1e-8	15	1.7e5	9.7e4	2.5e5	1.7e5	15	3.6e6	1.9e6	5.4e6	3.6e6

Table 4: Shown are, for functions f_{121} - f_{130} and for a given target difference to the optimal function value Δf : the number of successful trials (#); the expected running time to surpass $f_{\text{opt}} + \Delta f$ (ERT, see Figure 1); the 10%-tile and 90%-tile of the bootstrap distribution of ERT; the average number of function evaluations in successful trials or, if none was successful, as last entry the median number of function evaluations to reach the best function value (RT_{succ}). If $f_{\text{opt}} + \Delta f$ was never reached, figures in *italics* denote the best achieved Δf -value of the median trial and the 10% and 90%-tile trial. Furthermore, N denotes the number of trials, and mFE denotes the maximum of number of function evaluations executed in one trial. See Figure 1 for the names of functions.