Benchmarking Real-Coded Genetic Algorithm on Noisy Black-Box Optimization Testbed

Thanh-Do Tran
Dept. of Control and Instrumentation Eng.
Korea Maritime University
Busan 606-791, Korea
tdtran@hhu.ac.kr

Gang-Gyoo Jin
Div. of Comp., Control, and Elec. Comm. Eng.
Korea Maritime University
Busan 606-791, Korea
ggjin@hhu.ac.kr

ABSTRACT

Originally, genetic algorithms were developed based on the binary representation of candidate solutions in which each conjectured solution is a fixed-length string of binary numbers; however, real-valued representation scheme is basically superior and frequently utilized in addressing hard optimization tasks, particularly for the optimization in continuous domains under a black-box scenario. In this paper, we implement a generational real-coded genetic algorithm (RCGA)—which is composed of tournament selection, arithmetical crossover, and adaptive-range mutation—with a multiple independent restarts mechanism and benchmark it on the BBOB-2010 noisy testbed. The maximum number of function evaluations for each run is set to 50000 times the search space dimension. For 40-dimensional search space, the algorithm shows promising results with 6 functions being solved up to the precision of 10^{-8}.

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, Experimentation

Keywords

Benchmarking, Black-box optimization, Evolutionary computation, Real-coded genetic algorithm

1. INTRODUCTION

Real-coded genetic algorithms (RCGAs) have existed in a great deal of variants developed by practitioners of genetic algorithms [2, 5]. Each of which has its own merits accompanied by some specific type of problems. A general implementation of RCGA taking advantages of tournament selection, arithmetical crossover, and non-uniform mutation is in common use. Surprisingly, such an instance of RCGA is rarely come across being fully benchmarked on a set of noisy functions with diverse characteristics to examine when and how it is advantageous. The aim of this paper is to perform such tasks.

Detailed descriptions of the algorithm as well as the implementation of the RCGA in use have been reported in the complement of this paper [6]. In [6], a multiple independent restarts mechanism is incorporated into the RCGA using adaptive-range non-uniform mutation to benchmark the BBOB-2010 noiseless testbed. In the same manner, the exactly identical algorithm and parameter settings are utilized in this paper to tackle the BBOB-2010 noisy testbed. All details regarding the algorithm as well as the settings of parameters can be found in [6].

2. RESULTS AND DISCUSSION

Results from the experiments according to [3] on the benchmark functions given in [1, 4] are presented in Figures 1, 2, 3 and in Tables 1, 2 and 3.

It is observed that performance of the RCGA on the noisy testbed is mediocre and generally of secondary significance. The obtained results from Figure 1 show that the RCGA is able to solve functions f_{101}, f_{102}, f_{103}, f_{107}, f_{109}, f_{130} in 40-D. In case of 20-D, one more function, f_{128}, is solved. In 5-D, almost all functions can be solved with the precision as low as 10^{-2} to 10^{-1} as shown in Table 1 and Table 2. These results are trivial, however the applicability of the RCGA is still promising thanks to the simplicity of the algorithm as well as the ease of implementation. Moreover, there is still enough room for the integration of other enhancement techniques to improve the algorithm's performance. Further investigations such as adaptive adjustment of the control parameters and/or the utilization of hybrid schemes for crossover and mutation may bring better results.

3. REFERENCES

[1] S. Finck, N. Hansen, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2010: Presentation of the noisy functions. Technical Report 2009/21, Research Center PPE, 2010.

[2] M. Gen and R. Cheng. *Genetic Algorithms and Engineering Designs.* John Wiley and Sons, 1997.

[3] N. Hansen, A. Auger, S. Finck, and R. Ros. Real-parameter black-box optimization benchmarking

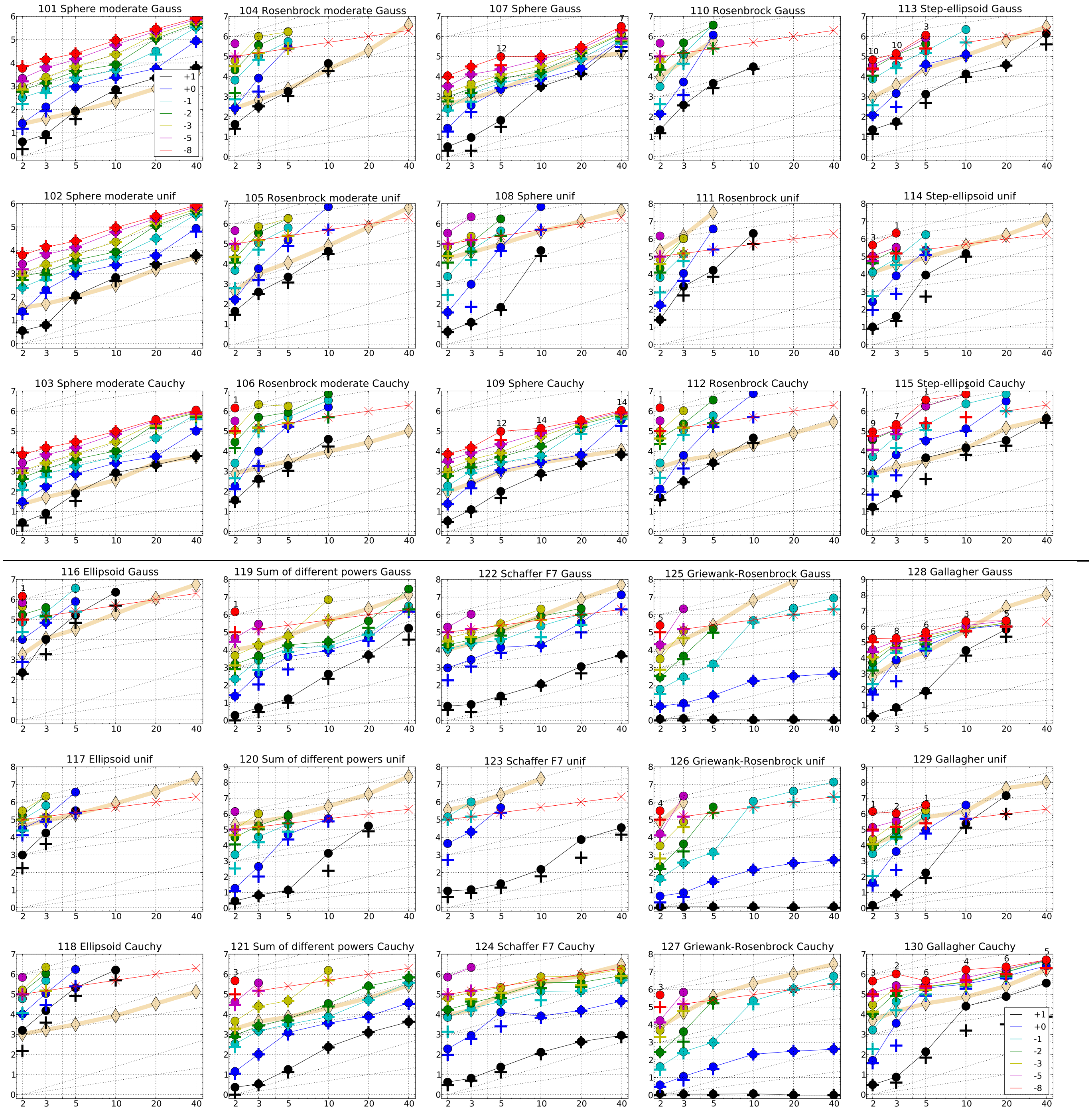

Figure 1: Expected Running Time (ERT, $\bullet$) to reach $f_{\rm opt} + \Delta f$ and median number of f-evaluations from successful trials ($+$), for $\Delta f = 10^{\{+1,0,-1,-2,-3,-5,-8\}}$ (the exponent is given in the legend of f_{101} and f_{130}) versus dimension in log-log presentation. For each function and dimension, $\mathrm{ERT}(\Delta f)$ equals to $\#\mathrm{FEs}(\Delta f)$ divided by the number of successful trials, where a trial is successful if $f_{\rm opt} + \Delta f$ was surpassed. The $\#\mathrm{FEs}(\Delta f)$ are the total number (sum) of f-evaluations while $f_{\rm opt} + \Delta f$ was not surpassed in the trial, from all (successful and unsuccessful) trials, and $f_{\rm opt}$ is the optimal function value. Crosses ($\times$) indicate the total number of f-evaluations, $\#\mathrm{FEs}(-\infty)$, divided by the number of trials. Numbers above ERT-symbols indicate the number of successful trials. Y-axis annotations are decimal logarithms. The thick light line with diamonds shows the single best results from BBOB-2009 for $\Delta f = 10^{-8}$. Additional grid lines show linear and quadratic scaling.

Δf	f_{101} in 5-D, N=15, mFE=30200					f_{101} in 20-D, N=15, mFE=290700				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	8.5e1	2.1e1	1.1e2	8.5e1	15	2.3e3	1.5e3	2.7e3	2.3e3
1	15	9.2e2	3.9e2	1.7e3	9.2e2	15	5.7e3	5.0e3	6.9e3	5.7e3
1e−1	15	2.1e3	1.0e3	2.8e3	2.1e3	15	3.2e4	1.0e4	5.8e4	3.2e4
1e−3	15	7.0e3	5.9e3	8.5e3	7.0e3	15	1.7e5	1.6e5	1.8e5	1.7e5
1e−5	15	1.4e4	1.0e4	1.6e4	1.4e4	15	2.3e5	2.3e5	2.4e5	2.3e5
1e−8	15	2.5e4	2.1e4	3.0e4	2.5e4	15	2.9e5	2.8e5	2.9e5	2.9e5

Δf	f_{102} in 5-D, N=15, mFE=31100					f_{102} in 20-D, N=15, mFE=293000				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.1e2	1.0e0	2.4e2	1.1e2	15	2.4e3	1.6e3	3.1e3	2.4e3
1	15	1.0e3	4.4e2	1.7e3	1.0e3	15	5.9e3	4.9e3	6.7e3	5.9e3
1e−1	15	2.0e3	9.4e2	2.8e3	2.0e3	15	3.3e4	1.5e4	6.1e4	3.3e4
1e−3	15	6.4e3	4.8e3	8.4e3	6.4e3	15	1.6e5	1.5e5	1.8e5	1.6e5
1e−5	15	1.3e4	9.8e3	1.7e4	1.3e4	15	2.3e5	2.2e5	2.4e5	2.3e5
1e−8	15	2.5e4	2.1e4	3.1e4	2.5e4	15	2.9e5	2.9e5	2.9e5	2.9e5

Δf	f_{103} in 5-D, N=15, mFE=34500					f_{103} in 20-D, N=15, mFE=979800				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	7.8e1	4.0e0	1.2e2	7.8e1	15	2.1e3	1.4e3	2.8e3	2.1e3
1	15	7.4e2	2.3e2	1.0e3	7.4e2	15	5.5e3	5.0e3	5.9e3	5.5e3
1e−1	15	1.8e3	8.0e2	2.9e3	1.8e3	15	4.5e4	9.9e3	8.1e4	4.5e4
1e−3	15	7.3e3	5.0e3	9.4e3	7.3e3	15	2.8e5	1.7e5	1.9e5	2.8e5
1e−5	15	1.5e4	1.2e4	2.0e4	1.5e4	15	3.4e5	2.4e5	9.6e5	3.4e5
1e−8	15	2.9e4	2.6e4	3.2e4	2.9e4	15	3.8e5	2.9e5	9.8e5	3.8e5

Δf	f_{104} in 5-D, N=15, mFE=250000					f_{104} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.8e3	7.6e2	2.5e3	1.8e3	0	*18e+0*	*18e+0*	*18e+0*	2.4e5
1	7	2.9e5	5.0e3	7.6e5	8.8e3	.	.	.	.	.
1e−1	5	5.6e5	1.3e4	1.3e6	5.5e4	.	.	.	.	.
1e−3	2	1.7e6	2.0e5	3.2e6	7.9e4	.	.	.	.	.
1e−5	0	*12e−1*	*29e−5*	*24e−1*	1.1e5	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{105} in 5-D, N=15, mFE=250000					f_{105} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	2.2e3	7.8e2	5.5e3	2.2e3	0	*18e+0*	*18e+0*	*18e+0*	2.2e5
1	11	1.5e5	8.8e3	4.6e5	6.2e4	.	.	.	.	.
1e−1	5	6.2e5	1.0e5	1.5e6	1.2e5	.	.	.	.	.
1e−3	2	1.8e6	2.5e5	4.0e6	2.1e5	.	.	.	.	.
1e−5	0	*51e−2*	*70e−5*	*17e−1*	1.1e5	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{106} in 5-D, N=15, mFE=250000					f_{106} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	2.0e3	7.4e2	4.7e3	2.0e3	0	*17e+0*	*17e+0*	*17e+0*	3.3e5
1	10	1.9e5	2.9e3	5.0e5	6.7e4	.	.	.	.	.
1e−1	6	5.0e5	2.8e4	1.2e6	1.3e5	.	.	.	.	.
1e−3	2	1.8e6	2.5e5	3.4e6	1.4e5	.	.	.	.	.
1e−5	0	*44e−2*	*75e−5*	*22e−1*	1.3e5	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{107} in 5-D, N=15, mFE=250000					f_{107} in 20-D, N=15, mFE=305400				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	6.5e1	8.0e0	1.9e2	6.5e1	15	1.5e4	6.8e3	2.4e4	1.5e4
1	15	2.4e3	3.3e2	3.9e3	2.4e3	15	2.6e4	1.9e4	3.7e4	2.6e4
1e−1	15	4.7e3	1.8e3	6.4e3	4.7e3	15	7.7e4	6.9e4	8.9e4	7.7e4
1e−3	15	1.2e4	8.9e3	1.6e4	1.2e4	15	1.9e5	1.8e5	1.9e5	1.9e5
1e−5	15	2.3e4	1.8e4	2.5e4	2.3e4	15	2.5e5	2.4e5	2.5e5	2.5e5
1e−8	12	9.8e4	3.4e4	2.9e5	3.6e4	15	3.0e5	2.9e5	3.0e5	3.0e5

Δf	f_{108} in 5-D, N=15, mFE=250000					f_{108} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	6.9e1	1.3e1	2.0e2	6.9e1	0	*22e+0*	*14e+0*	*28e+0*	3.1e5
1	14	6.7e4	4.5e3	1.5e5	4.9e4	.	.	.	.	.
1e−1	6	4.4e5	5.4e4	9.4e5	6.4e4	.	.	.	.	.
1e−3	0	*16e−2*	*90e−4*	*91e−2*	1.2e5	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{109} in 5-D, N=15, mFE=250000					f_{109} in 20-D, N=15, mFE=962800				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	9.9e1	1.4e1	7.2e2	9.9e1	15	2.5e3	1.8e3	3.5e3	2.5e3
1	15	1.1e3	4.3e2	1.7e3	1.1e3	15	6.6e3	5.4e3	7.5e3	6.6e3
1e−1	15	2.7e3	7.2e2	3.6e3	2.7e3	15	1.3e5	4.5e4	1.0e5	1.3e5
1e−3	15	9.0e3	6.8e3	1.2e4	9.0e3	15	2.5e5	1.9e5	3.8e5	2.5e5
1e−5	15	2.2e4	1.6e4	2.7e4	2.2e4	15	3.1e5	2.5e5	6.7e5	3.1e5
1e−8	12	9.7e4	3.2e4	2.9e5	3.5e4	15	3.6e5	3.0e5	9.6e5	3.6e5

Δf	f_{110} in 5-D, N=15, mFE=250000					f_{110} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	4.6e3	1.0e3	1.5e4	4.6e3	0	*20e+0*	*20e+0*	*21e+0*	1.7e5
1	3	1.2e6	1.5e5	2.3e6	1.6e5	.	.	.	.	.
1e−1	1	3.6e6	4.0e5	8.9e6	1.5e5	.	.	.	.	.
1e−3	0	*18e−1*	*12e−2*	*28e−1*	1.8e5	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{111} in 5-D, N=15, mFE=250000					f_{111} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.6e4	1.7e3	3.6e4	1.6e4	0	*49e+0*	*33e+0*	*85e+0*	1.2e5
1	1	3.7e6	4.3e5	7.3e6	1.8e5	.	.	.	.	.
1e−1	0	*26e−1*	*11e−1*	*33e−1*	1.7e5	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{112} in 5-D, N=15, mFE=250000					f_{112} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	2.7e3	8.4e2	4.1e3	2.7e3	0	*18e+0*	*17e+0*	*18e+0*	3.2e5
1	10	2.2e5	1.9e4	4.7e5	9.3e4	.	.	.	.	.
1e−1	5	5.9e5	4.0e4	1.4e6	9.5e4	.	.	.	.	.
1e−3	0	*70e−2*	*14e−3*	*21e−1*	1.8e5	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{113} in 5-D, N=15, mFE=250000					f_{113} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.3e3	2.9e1	3.7e3	1.3e3	15	3.9e4	2.4e4	5.8e4	3.9e4
1	15	4.1e4	4.4e3	1.2e5	4.1e4	0	*23e−1*	*13e−1*	*28e−1*	1.6e5
1e−1	12	1.8e5	6.8e4	4.2e5	1.2e5	.	.	.	.	.
1e−3	4	8.4e5	1.1e5	2.2e6	1.5e5	.	.	.	.	.
1e−5	4	8.4e5	1.3e5	1.7e6	1.5e5	.	.	.	.	.
1e−8	3	1.2e6	1.7e5	2.8e6	1.7e5	.	.	.	.	.

Δf	f_{114} in 5-D, N=15, mFE=250000					f_{114} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	9.0e3	2.5e1	2.9e4	9.0e3	0	*46e+0*	*24e+0*	*11e+1*	1.6e5
1	9	2.3e5	4.1e4	5.3e5	6.4e4	.	.	.	.	.
1e−1	2	1.8e6	1.4e5	4.9e6	1.4e5	.	.	.	.	.
1e−3	0	*41e−2*	*67e−3*	*35e−1*	1.3e5	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{115} in 5-D, N=15, mFE=250000					f_{115} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	4.7e3	1.9e2	3.0e3	4.7e3	15	3.5e4	1.1e4	4.1e4	3.5e4
1	15	3.3e4	2.2e3	6.1e4	3.3e4	4	3.1e6	1.5e5	8.8e6	4.0e5
1e−1	11	2.0e5	4.7e4	5.5e5	1.1e5	2	7.1e6	9.6e5	1.5e7	5.5e5
1e−3	2	1.8e6	1.9e5	4.8e6	1.7e5	0	*11e−1*	*91e−3*	*25e−1*	3.9e5
1e−5	2	1.8e6	1.4e5	4.1e6	1.7e5	.	.	.	.	.
1e−8	1	3.6e6	3.9e5	9.4e6	1.4e5	.	.	.	.	.

Δf	f_{116} in 5-D, N=15, mFE=250000					f_{116} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	11	1.7e5	3.7e4	3.1e5	7.6e4	0	*22e+1*	*98e+0*	*54e+1*	2.1e5
1	4	7.9e5	9.1e4	2.1e6	1.0e5	.	.	.	.	.
1e−1	1	3.6e6	6.0e5	9.3e6	9.6e4	.	.	.	.	.
1e−3	0	*22e−1*	*11e−2*	*37e+0*	1.4e5	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{117} in 5-D, N=15, mFE=250000					f_{117} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	8	3.2e5	2.9e4	6.3e5	1.0e5	0	*14e+2*	*50e+1*	*38e+2*	3.4e5
1	1	3.7e6	6.5e5	8.4e6	1.5e5	.	.	.	.	.
1e−1	0	*45e−1*	*13e−1*	*76e+0*	1.3e5	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{118} in 5-D, N=15, mFE=250000					f_{118} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	9	2.1e5	1.1e4	6.9e5	4.5e4	0	*10e+1*	*66e+0*	*19e+1*	3.3e5
1	2	1.7e6	2.0e5	4.6e6	1.1e5	.	.	.	.	.
1e−1	0	*45e−1*	*16e−2*	*47e+0*	1.6e5	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{119} in 5-D, N=15, mFE=250000					f_{119} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.7e1	4.0e0	3.9e1	1.7e1	15	5.0e3	2.2e3	9.4e3	5.0e3
1	15	4.1e3	3.0e2	1.1e4	4.1e3	15	4.8e4	1.9e4	8.3e4	4.8e4
1e−1	15	1.3e4	2.5e3	2.1e4	1.3e4	15	8.6e4	6.0e4	9.7e4	8.6e4
1e−3	15	6.4e4	4.4e4	1.0e5	6.4e4	0	*57e−4*	*27e−4*	*15e−3*	2.4e5
1e−5	0	*20e−5*	*43e−6*	*42e−5*	1.2e5	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	f_{120} in 5-D, N=15, mFE=250000					f_{120} in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.4e1	1.0e0	3.0e1	1.4e1	15	1.7e5	1.3e4	7.9e5	1.7e5
1	14	5.0e4	3.1e2	1.6e5	3.2e4	0	*61e−1*	*31e−1*	*85e−1*	4.5e5
1e−1	9	2.2e5	2.2e4	5.5e5	5.4e4	.	.	.	.	.
1e−3	0	*50e−3*	*31e−4*	*75e−2*	8.8e4	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Table 1: Shown are, for functions f_{101}-f_{120} and for a given target difference to the optimal function value Δf: the number of successful trials (#); the expected running time to surpass $f_{\rm opt} + \Delta f$ (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 $f_{\rm opt} + \Delta f$ 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.

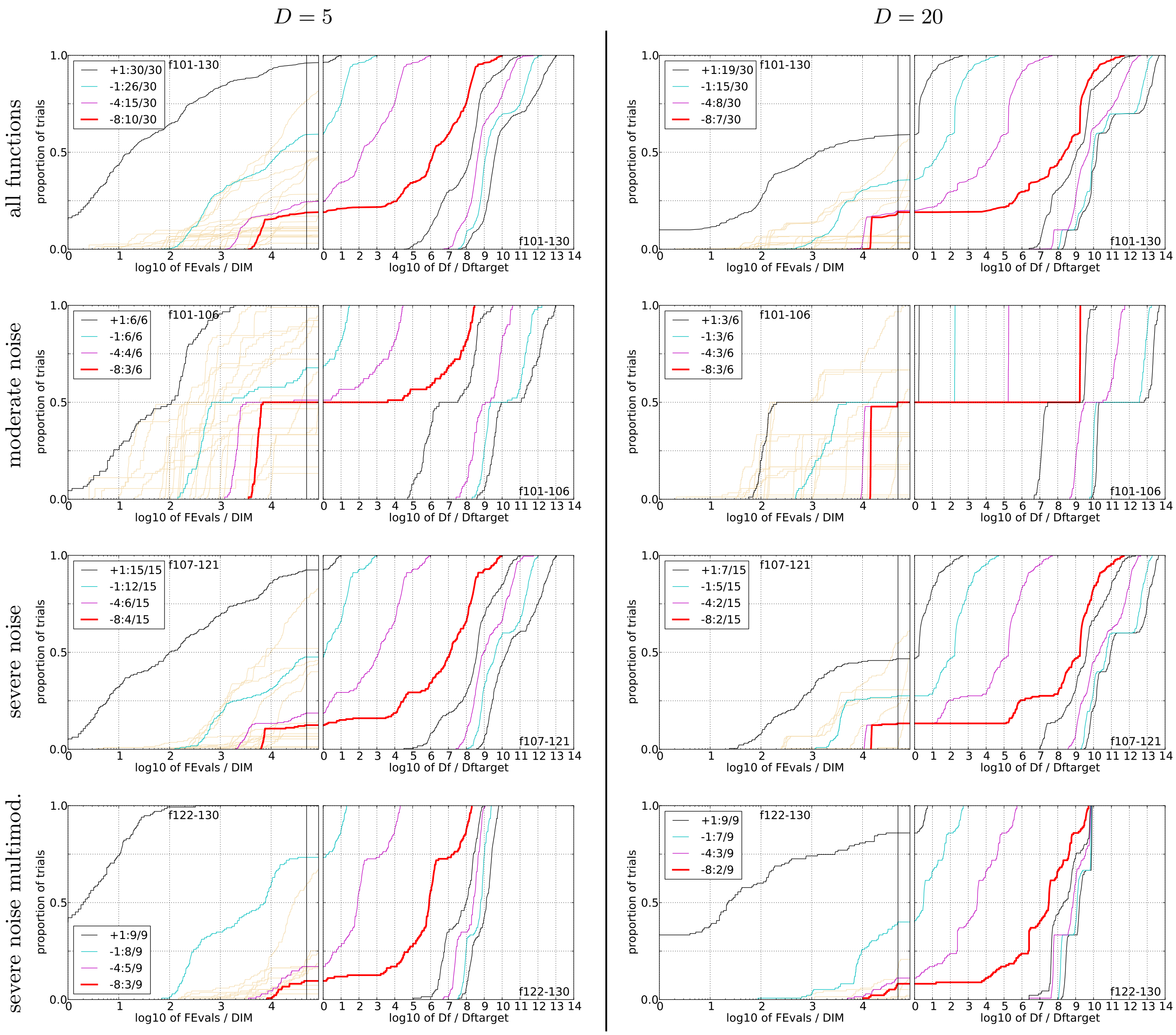

Figure 2: Empirical cumulative distribution functions (ECDFs), plotting the fraction of trials versus running time (left subplots) or versus Δf (right subplots). The thick red line represents the best achieved results. Left subplots: ECDF of the running time (number of function evaluations), divided by search space dimension D, to fall below $f_{\rm opt} + \Delta f$ 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 $D, 10\,D, 100\,D\ldots$ function evaluations (from right to left cycling black-cyan-magenta). 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. Light brown lines in the background show ECDFs for target value 10^{-8} of all algorithms benchmarked during BBOB-2009.

Δf	*f*121 in 5-D, N=15, mFE=250000					*f*121 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.8e1	2.0e0	3.6e1	1.8e1	15	1.2e3	6.9e2	1.8e3	1.2e3
1	15	1.3e3	4.2e2	2.9e3	1.3e3	15	8.0e3	6.4e3	1.0e4	8.0e3
1e−1	15	3.1e3	1.9e3	3.9e3	3.1e3	15	5.1e4	2.4e4	7.0e4	5.1e4
1e−3	15	4.8e4	8.3e3	1.5e5	4.8e4	0	*41e−4*	*18e−4*	*53e−4*	3.3e5
1e−5	0	*22e−5*	*12e−5*	*45e−5*	5.5e4	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*123 in 5-D, N=15, mFE=250000					*f*123 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	2.3e1	2.5e0	6.0e1	2.3e1	15	7.4e3	1.9e2	1.6e4	7.4e3
1	6	4.8e5	6.0e4	8.1e5	1.0e5	0	*48e−1*	*39e−1*	*59e−1*	5.4e5
1e−1	0	*11e−1*	*41e−2*	*19e−1*	1.5e5	.	.	.	.	.
1e−3	.	.	.	.	.	.	.	.	.	.
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*125 in 5-D, N=15, mFE=250000					*f*125 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.1e0	1.0e0	2.0e0	1.1e0	15	1.1e0	1.0e0	2.0e0	1.1e0
1	15	2.6e1	6.0e0	5.1e1	2.6e1	15	3.2e2	2.2e2	3.8e2	3.2e2
1e−1	15	1.6e3	8.8e2	3.2e3	1.6e3	5	2.3e6	1.7e5	4.8e6	3.4e5
1e−3	0	*82e−4*	*30e−4*	*15e−3*	1.2e5	0	*33e−2*	*25e−3*	*34e−2*	3.9e5
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*127 in 5-D, N=15, mFE=250000					*f*127 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.1e0	1.0e0	2.0e0	1.1e0	15	1.0e0	1.0e0	1.0e0	1.0e0
1	15	4.1e1	3.0e0	7.9e1	4.1e1	15	3.1e2	2.5e2	3.5e2	3.1e2
1e−1	15	9.9e2	5.5e2	1.3e3	9.9e2	9	1.1e6	1.7e3	2.3e6	4.1e5
1e−3	0	*95e−4*	*57e−4*	*17e−3*	1.1e5	0	*59e−3*	*25e−3*	*24e−2*	9.9e5
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*129 in 5-D, N=15, mFE=250000					*f*129 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.8e2	7.0e0	2.1e2	1.8e2	1	1.5e7	1.1e6	3.6e7	5.8e5
1	14	9.1e4	5.6e3	2.2e5	7.3e4	0	*34e+0*	*15e+0*	*53e+0*	5.1e5
1e−1	4	7.6e5	7.4e4	1.6e6	7.1e4	.	.	.	.	.
1e−3	2	1.7e6	1.2e5	3.3e6	1.0e5	.	.	.	.	.
1e−5	1	3.7e6	6.9e5	7.4e6	1.9e5	.	.	.	.	.
1e−8	1	3.7e6	4.5e5	6.7e6	2.0e5	.	.	.	.	.

Δf	*f*122 in 5-D, N=15, mFE=250000					*f*122 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	2.4e1	3.0e0	5.3e1	2.4e1	15	1.1e3	1.5e2	3.6e3	1.1e3
1	15	1.4e4	8.4e2	3.8e4	1.4e4	12	3.5e5	6.4e4	1.1e6	9.7e4
1e−1	15	3.4e4	1.1e4	5.3e4	3.4e4	8	1.1e6	1.7e5	3.2e6	2.0e5
1e−3	8	3.1e5	5.9e4	8.2e5	8.7e4	0	*98e−3*	*18e−4*	*13e−1*	2.7e5
1e−5	0	*72e−5*	*24e−6*	*12e−3*	7.4e4	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*124 in 5-D, N=15, mFE=250000					*f*124 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	2.4e1	2.0e0	4.7e1	2.4e1	15	4.4e2	1.4e2	8.0e2	4.4e2
1	15	1.3e4	8.1e2	4.4e4	1.3e4	15	1.6e4	1.1e4	2.1e4	1.6e4
1e−1	15	4.2e4	1.0e4	6.1e4	4.2e4	15	1.5e5	1.4e5	1.6e5	1.5e5
1e−3	10	2.2e5	6.9e4	6.2e5	9.1e4	10	7.8e5	2.7e5	2.3e6	2.8e5
1e−5	0	*55e−5*	*35e−6*	*76e−4*	8.3e4	0	*39e−5*	*77e−6*	*23e−3*	3.4e5
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*126 in 5-D, N=15, mFE=250000					*f*126 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.1e0	1.0e0	2.0e0	1.1e0	15	1.1e0	1.0e0	1.0e0	1.1e0
1	15	3.3e1	2.0e0	5.9e1	3.3e1	15	3.4e2	2.6e2	4.2e2	3.4e2
1e−1	15	1.4e3	3.5e2	2.1e3	1.4e3	3	4.2e6	7.4e4	9.7e6	1.8e5
1e−3	0	*11e−3*	*35e−4*	*20e−3*	1.4e5	0	*32e−2*	*25e−3*	*39e−2*	4.1e5
1e−5	.	.	.	.	.	.	.	.	.	.
1e−8	.	.	.	.	.	.	.	.	.	.

Δf	*f*128 in 5-D, N=15, mFE=250000					*f*128 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	7.7e1	7.0e0	1.4e2	7.7e1	10	6.7e5	5.9e4	1.4e6	1.7e5
1	14	5.6e4	1.1e3	2.0e5	3.8e4	7	1.3e6	1.3e5	3.4e6	2.0e5
1e−1	11	1.2e5	6.6e3	3.0e5	3.2e4	6	1.7e6	5.7e4	4.3e6	2.1e5
1e−3	8	2.7e5	1.3e4	8.1e5	4.7e4	6	1.7e6	1.1e5	4.3e6	2.5e5
1e−5	8	2.8e5	3.3e4	8.0e5	6.5e4	5	2.3e6	1.8e5	5.8e6	3.1e5
1e−8	6	4.5e5	5.3e4	8.5e5	7.2e4	5	2.4e6	2.4e5	4.6e6	3.7e5

Δf	*f*130 in 5-D, N=15, mFE=250000					*f*130 in 20-D, N=15, mFE=1.00e6				
	#	ERT	10%	90%	RT_{succ}	#	ERT	10%	90%	RT_{succ}
10	15	1.4e2	1.1e1	4.0e2	1.4e2	15	7.8e4	2.2e3	3.4e5	7.8e4
1	9	2.0e5	7.0e1	5.7e5	3.6e4	10	8.1e5	1.0e4	2.0e6	3.1e5
1e−1	9	2.3e5	2.4e3	5.4e5	5.9e4	8	1.3e6	2.0e4	2.8e6	3.9e5
1e−3	9	2.5e5	3.2e4	5.6e5	7.8e4	7	1.7e6	3.4e4	3.5e6	5.2e5
1e−5	8	3.2e5	5.4e4	8.0e5	9.7e4	7	1.8e6	5.5e5	4.6e6	7.0e5
1e−8	6	4.8e5	6.4e4	8.2e5	1.0e5	6	2.3e6	8.7e5	4.9e6	8.3e5

Table 2: 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 $f_{\mathrm{opt}} + \Delta f$ (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 $f_{\mathrm{opt}} + \Delta f$ 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.**

Table 3: ERT **loss ratio (see Figure 3) compared to the respective best result from BBOB-2009 for budgets given in the first column. The last row RL_{US}/D gives the number of function evaluations in unsuccessful runs divided by dimension. Shown are the smallest, 10%-tile, 25%-tile, 50%-tile, 75%-tile and 90%-tile value (smaller values are better).**

	*f*101–*f*130 in 5-D, maxFE/D=50000					
#FEs/D	best	10%	25%	**med**	75%	90%
2	0.56	1.0	1.6	2.5	6.3	10
10	0.94	1.1	2.1	4.2	5.8	28
100	2.5	5.1	7.0	9.2	15	2.6e2
1e3	2.4	7.5	12	36	79	2.6e3
1e4	1.8	11	19	43	1.6e2	1.1e4
1e5	0.42	4.9	17	27	2.4e2	3.7e2
RL_{US}/D	5e4	5e4	5e4	5e4	5e4	5e4

	*f*101–*f*130 in 20-D, maxFE/D=50000					
#FEs/D	best	10%	25%	**med**	75%	90%
2	1.0	1.0	1.9	32	40	40
10	0.42	1.1	2.9	35	2.0e2	2.0e2
100	0.42	0.88	1.6	7.6	24	1.0e3
1e3	0.48	0.83	2.1	12	37	2.0e4
1e4	0.93	2.2	4.4	36	1.9e2	1.0e5
1e5	0.10	4.1	10	58	5.4e2	1.0e6
1e6	0.14	3.2	64	1.6e2	4.9e3	2.1e6
RL_{US}/D	5e4	5e4	5e4	5e4	5e4	5e4

2010: Experimental setup. Technical Report RR-7215, INRIA, 2010.

[4] 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. Updated February 2010.

[5] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*. Springer, 1996.

[6] T.-D. Tran and G.-G. Jin. Real-coded genetic algorithm benchmarked on noiseless black-box optimization testbed. In *Workshop Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2010)*, 2010.

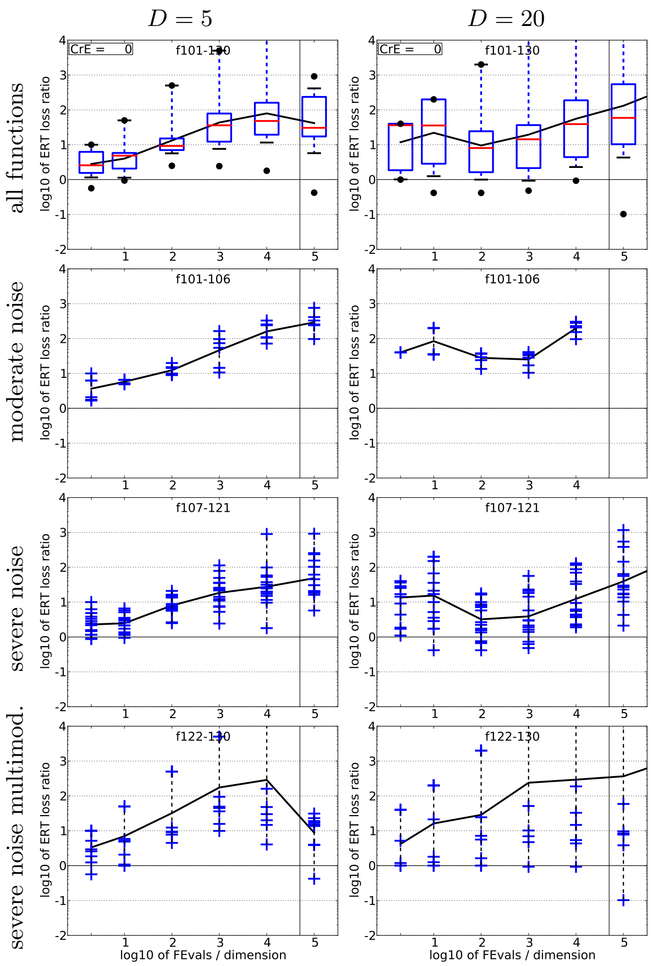

Figure 3: ERT loss ratio versus given budget FEvals. The target value f_t for ERT (see Figure 1) is the smallest (best) recorded function value such that $\mathrm{ERT}(f_t) \le \mathrm{FEvals}$ for the presented algorithm. Shown is FEvals divided by the respective best $\mathrm{ERT}(f_t)$ from BBOB-2009 for functions f_{101}–f_{130} in 5-D and 20-D. Each ERT is multiplied by $\exp(\mathrm{CrE})$ correcting for the parameter crafting effort. Line: geometric mean. Box-Whisker error bar: 25-75%-tile with median (box), 10-90%-tile (caps), and minimum and maximum ERT loss ratio (points). The vertical line gives the maximal number of function evaluations in this function subset.