

Comparison of Cauchy EDA and BIPOP-CMA-ES Algorithms on the BBOB Noiseless Testbed

Petr Pošík
Czech Technical University in Prague
Faculty of Electrical Engineering, Department of Cybernetics
Technická 2, 166 27 Prague 6, Czech Republic
posik@labe.felk.cvut.cz

ABSTRACT

Estimation-of-distribution algorithm using Cauchy sampling distribution is compared with the bi-population CMA evolutionary strategy which was one of the best contenders in the black-box optimization benchmarking workshop in 2009. The results clearly indicate that the CMA evolutionary strategy is in all respects a better optimization algorithm than the Cauchy estimation-of-distribution algorithm. This paper compares both algorithms in more detail and adds to the understanding of their key features and differences.

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, Estimation-of-distribution algorithm, Cauchy distribution, Evolutionary strategy, Covariance matrix adaptation

1. INTRODUCTION

The 2010 issue of the black-box optimization benchmarking methodology (BBOB) [3] allows for a detailed comparison of 2 algorithms on the BBOB functions testbed. In this article, two algorithms benchmarked during the BBOB 2009 workshop are further compared. Data for both algorithms were taken from 2009 benchmarking, but the comparison is made using the new BBOB 2010 post-processing scripts and templates. Both algorithms fall into the class of evolutionary optimization algorithms and both algorithms use unimodal

distribution as a mean for generating new offspring; however, there are several important differences between them and this paper clearly shows which algorithm is better. The two algorithms selected for the comparison are:

- The bi-population variant of the evolutionary strategy with covariance matrix adaptation (BIPOP-CMA-ES) [2] which belongs to the best algorithms of the BBOB 2009 comparison in terms of speed and success ratio, and thus was selected as the reference algorithm.
- The estimation-of-distribution algorithm (EDA) with Cauchy sampling distribution (Cauchy EDA) [5]. In BBOB 2010, further comparisons of the Cauchy EDA with other algorithms (G3PCX [6] and Rosenbrock's algorithm [7]) are planned, and this article anchors the relative performances of the respective algorithm pairs to one of the best algorithms in BBOB 2009.

In the next section, both algorithms are shortly described and their differences are emphasized. Sec. 3 contains all the results used to compare the algorithms and their discussions. After the presentation of the time demands of both algorithms in Sec. 4, Sec. 5 concludes the paper.

2. ALGORITHM PRESENTATION

The exact descriptions of the algorithms along with the parameter settings can be found in [5] and [2], respectively. Apart of the unimodality of the sampling distribution, the algorithms differ foremost in the following aspects:

- The probabilistic model used in BIPOP-CMA-ES is Gaussian, while the EDA uses Cauchy distribution.
- In BIPOP-CMA-ES, each generation, the Gaussian distribution is *updated* (the recent model parameters explicitly take part in the process of creating new values of model parameters), while in Cauchy EDA all the distribution parameters are *computed from scratch* (and thus a larger population is needed).
- The restart strategy of BIPOP-CMA-ES allows the algorithm to use different population sizes for each restart. In Cauchy EDA, the population size depends only on the dimensionality of the problem being solved. Very often, the BIPOP-CMA-ES population size is much smaller when compared to the population size of Cauchy EDA, which allows the BIPOP-CMA-ES algorithm to converge faster and to be restarted more often.

For both algorithms, the crafting effort CrE= 0.

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'10, July 7–11, 2010, Portland, Oregon, USA.
Copyright 2010 ACM 978-1-4503-0073-5/10/07 ...\$10.00.

Table 2: The average time demands per function evaluation (in microseconds) of the two compared algorithms.

| Dim | 2 | 3 | 5 | 10 | 20 | 40 |
|--------------|-----|-----|-----|-----|-----|-----|
| BIPOP-CMA-ES | 280 | 240 | 200 | 180 | 180 | 200 |
| CauchyEDA | 51 | 17 | 9 | 9 | 11 | NA |

3. RESULTS

Results from experiments according to [3] on the benchmark functions given in [1, 4] are presented in Figures 1, 2 and 3 and in Table 1. The **expected running time (ERT)**, used in the figures and table, depends on a given target function value, $f_t = f_{\text{opt}} + \Delta f$, and is computed over all relevant trials as the number of function evaluations executed during each trial while the best function value did not reach f_t , summed over all trials and divided by the number of trials that actually reached f_t [3, 8]. **Statistical significance** is tested with the rank-sum test for a given target Δf_t (10^{-8} in Figure 1) using, for each trial, either the number of needed function evaluations to reach Δf_t (inverted and multiplied by -1), or, if the target was not reached, the best Δf -value achieved, measured only up to the smallest number of overall function evaluations for any unsuccessful trial under consideration.

The most important observation that can be made from Figures 1, 2, and 3 and from Table 1 is that *BIPOP-CMA-ES is simply more reliable (has higher success rates) and typically 1-2 orders of magnitude faster than Cauchy EDA!* BIPOP-CMA-ES outperforms Cauchy EDA for all functions, (virtually) all dimensions and (virtually) all target levels. The few exceptions happen for low dimensional functions (2D or 3D) and for a very narrow range of target levels.

In Fig. 1, we can see rather regular behaviour of the ERT ratios for unimodal¹ functions (1, 2, 5–14). The ERT ratio is often almost constant (between 1 and 100) for a broad range of the target levels.

For multimodal functions (3, 4, 15–24), the Cauchy EDA algorithm does not work well. The long running times allow for a limited number of restarts only, and it is able to solve only low-dimensional versions of some of the multimodal benchmark functions.

4. CPU TIMING EXPERIMENTS

The time requirements of both algorithms are taken from the respective articles, [2] and [5]. The multistart algorithm was run with the maximal number of evaluations set to 10^5 , the basic algorithm was restarted for at least 30 seconds. These experiments have been conducted with an Intel dual core T5600 processor with 1.8 GHz under Linux 2.6.27-11 using MATLAB R2008a for BIPOP-CMA-ES, and on Intel Core 2 CPU, T5600, 1.83 GHz, 1 GB RAM with Windows XP SP3 in MATLAB R2007b for Cauchy EDA. The comparison of the average time demands per function evaluation are shown in Table 2.

The differences in the average time needed for function evaluation are caused by the different population sizes. While BIPOP-CMA-ES often uses populations of a few (or a few

tens of) individuals, Cauchy EDA needs larger populations which means that the evaluation routine is called less often and can take advantage of the MATLAB matrix processing capabilities to a larger extent.

5. CONCLUSIONS

The results indicate that BIPOP-CMA-ES clearly dominates the Cauchy EDA algorithm regardless of the particular optimization conditions. The adaptation scheme used in CMA-ES needs lower population sizes, is thus faster, and allows for more algorithm restarts. For the functions in the testbed, it seems to be better to have fast local optimizer with the possibility to restart it often.

Acknowledgements

The author is supported by the Grant Agency of the Czech Republic with the grant no. 102/08/P094 entitled “Machine learning methods for solution construction in evolutionary algorithms”.

6. REFERENCES

- [1] 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.
- [2] N. Hansen. Benchmarking a bi-population CMA-ES on the BBOB-2009 function testbed. In *GECCO (Companion)*, pages 2389–2396, 2009.
- [3] N. Hansen, A. Auger, S. Finck, and R. Ros. Real-parameter black-box optimization benchmarking 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: Noiseless functions definitions. Technical Report RR-6829, INRIA, 2009. Updated February 2010.
- [5] P. Pošík. BBOB-benchmarking a simple estimation-of-distribution algorithm with Cauchy distribution. In *GECCO '09: Proceedings of the 11th annual conference companion on Genetic and evolutionary computation conference*, pages 2309–2314, New York, NY, USA, 2009. ACM.
- [6] P. Pošík. BBOB-benchmarking the generalized generation gap model with parent centric crossover. In *GECCO '09: Proceedings of the 11th annual conference companion on Genetic and evolutionary computation conference*, pages 2321–2328, New York, NY, USA, 2009. ACM.
- [7] P. Pošík. BBOB-benchmarking the Rosenbrock’s local search algorithm. In *GECCO '09: Proceedings of the 11th annual conference companion on Genetic and evolutionary computation conference*, pages 2337–2342, New York, NY, USA, 2009. ACM.
- [8] K. Price. Differential evolution vs. the functions of the second ICEO. In *Proceedings of the IEEE International Congress on Evolutionary Computation*, pages 153–157, 1997.

¹The Rosenbrock’s functions are actually multimodal, but the local optimum does not pose many difficulties to optimization algorithms.

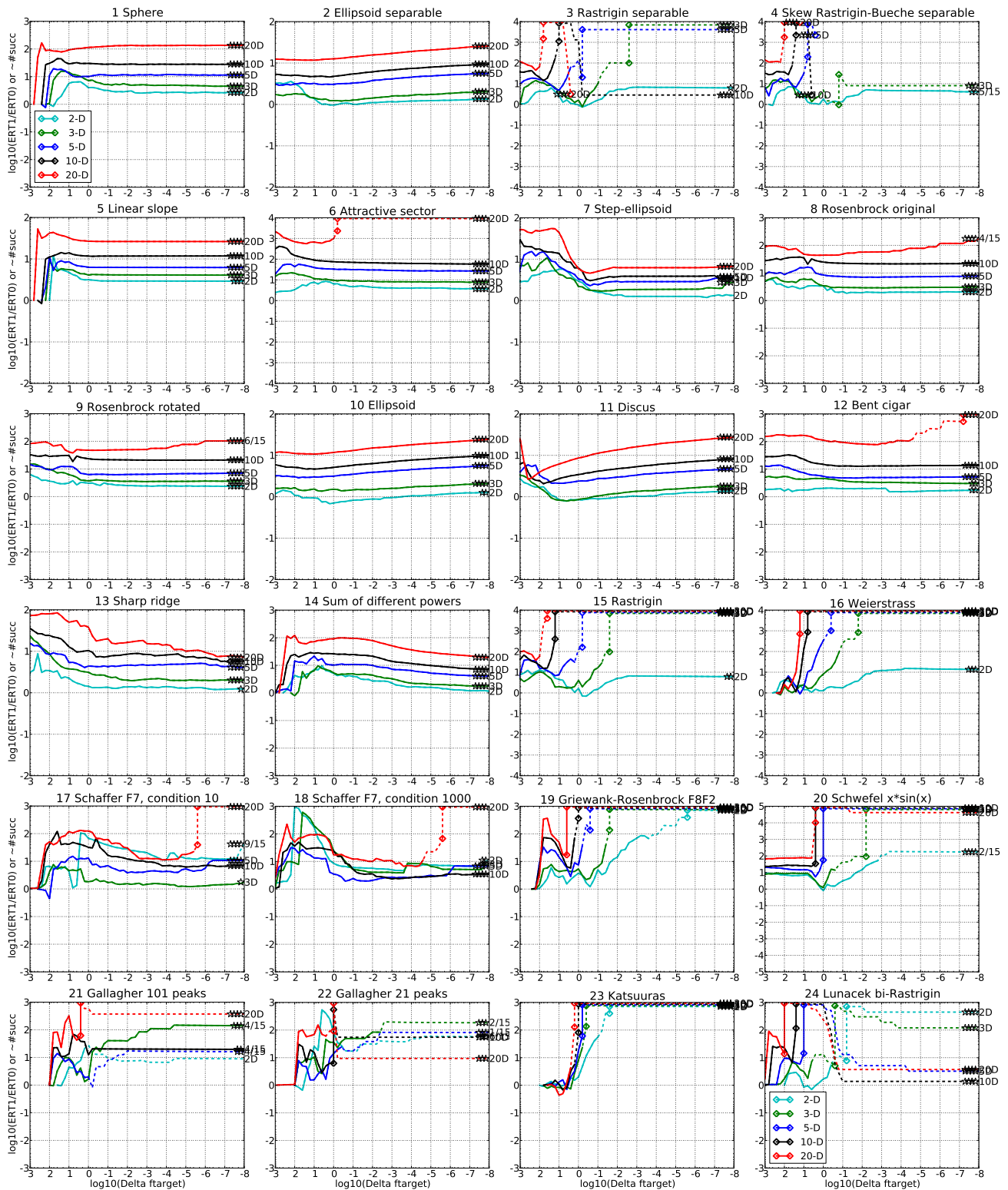


Figure 1: ERT ratio of CauchyEDA divided by BIPOP-CMA-ES versus $\log_{10}(\Delta f)$ for f_1 – f_{24} in 2, 3, 5, 10, 20, 40-D. Ratios $< 10^0$ indicate an advantage of CauchyEDA, smaller values are always better. The line gets dashed when for any algorithm the ERT exceeds thrice the median of the trial-wise overall number of f -evaluations for the same algorithm on this function. Symbols indicate the best achieved Δf -value of one algorithm (ERT gets undefined to the right). The dashed line continues as the fraction of successful trials of the other algorithm, where 0 means 0% and the y-axis limits mean 100%, values below zero for CauchyEDA. The line ends when no algorithm reaches Δf anymore. The number of successful trials is given, only if it was in $\{1 \dots 9\}$ for CauchyEDA (1st number) and non-zero for BIPOP-CMA-ES (2nd number). Results are significant with $p = 0.05$ for one star and $p = 10^{-\#\ast}$ otherwise, with Bonferroni correction within each figure.

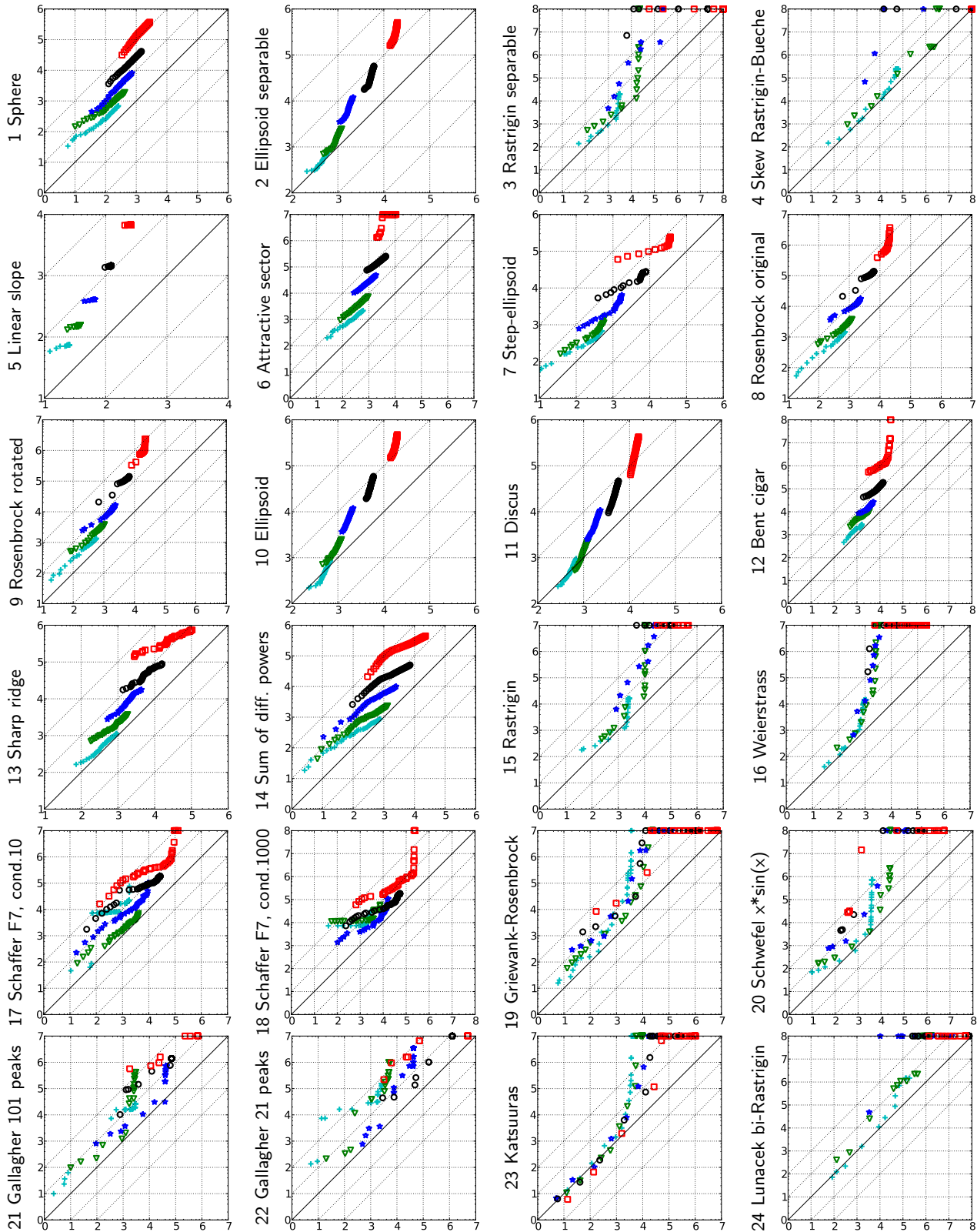


Figure 2: Expected running time (ERT in \log_{10} of number of function evaluations) of CauchyEDA versus BIPOP-CMA-ES for 46 target values $\Delta f \in [10^{-8}, 10]$ in each dimension for functions f_1 - f_{24} . Markers on the upper or right edge indicate that the target value was never reached by CauchyEDA or BIPOP-CMA-ES respectively. Markers represent dimension: 2:+, 3: ∇ , 5:*, 10: \circ , 20: \square , 40: \diamond .

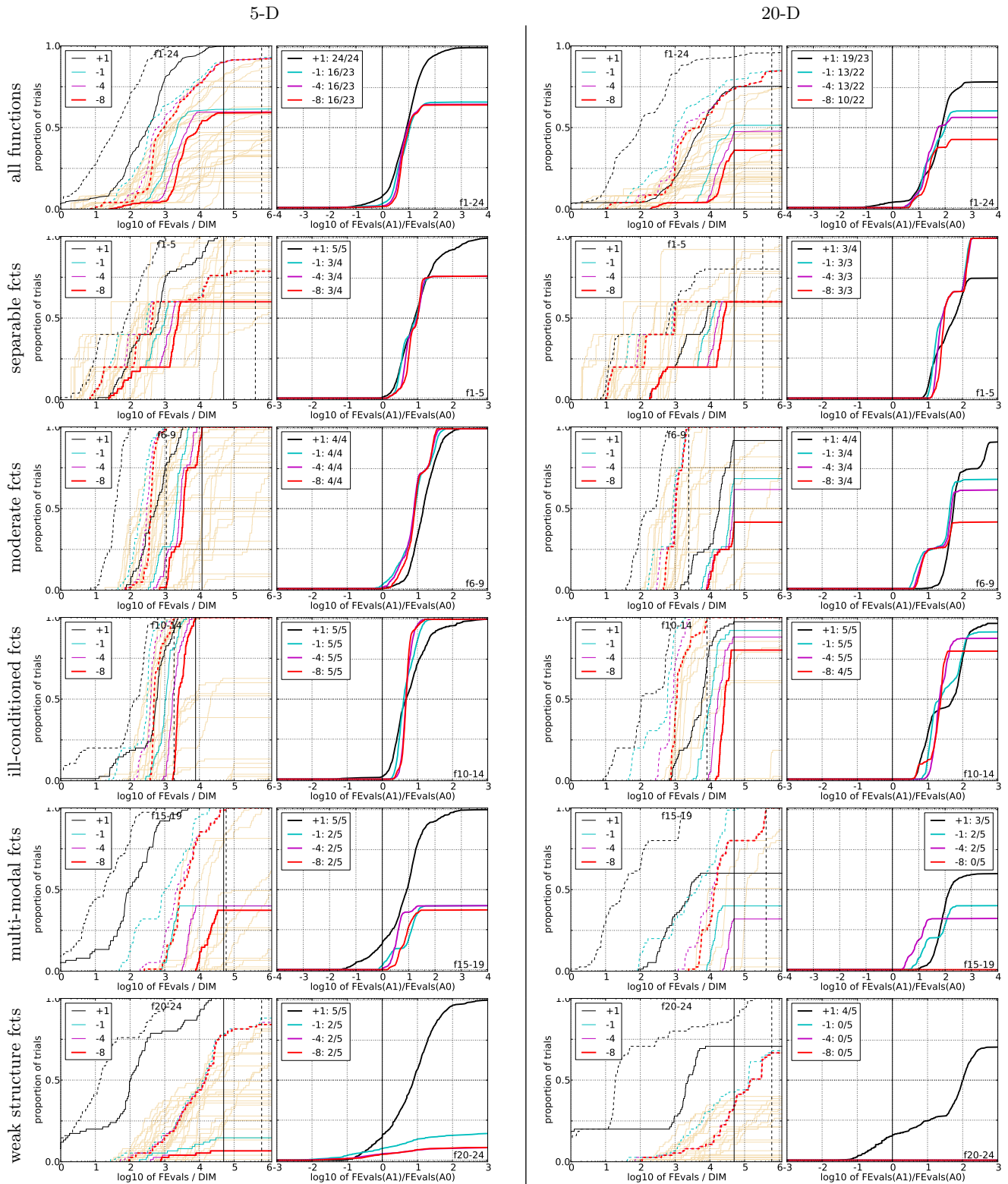


Figure 3: Empirical cumulative distributions (ECDF) of run lengths and speed-up ratios in 5-D (left) and 20-D (right). Left sub-columns: ECDF of the number of function evaluations divided by dimension D (FEvals/ D) to reach a target value $f_{\text{opt}} + \Delta f$ with $\Delta f = 10^k$, where $k \in \{1, -1, -4, -8\}$ is given by the first value in the legend, for CauchyEDA (solid) and BIPOP-CMA-ES (dashed). Light beige lines show the ECDF of FEvals for target value $\Delta f = 10^{-8}$ of algorithms benchmarked during BBOB-2009. Right sub-columns: ECDF of FEval ratios of CauchyEDA divided by BIPOP-CMA-ES, all trial pairs for each function. Pairs where both trials failed are disregarded, pairs where one trial failed are visible in the limits being > 0 or < 1 . The legends indicate the number of functions that were solved in at least one trial (CauchyEDA first).

