



HAL
open science

Adaptive Strategy Selection within Differential Evolution on the BBOB-2010 Noiseless Benchmark

Álvaro Fialho, Raymond Ros

► **To cite this version:**

Álvaro Fialho, Raymond Ros. Adaptive Strategy Selection within Differential Evolution on the BBOB-2010 Noiseless Benchmark. [Research Report] RR-7259, 2010. inria-00476160v1

HAL Id: inria-00476160

<https://inria.hal.science/inria-00476160v1>

Submitted on 23 Apr 2010 (v1), last revised 30 Apr 2010 (v2)

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.



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

*Analysis of Adaptive Strategy Selection within
Differential Evolution on the BBOB-2010 Noiseless
Benchmark*

Álvaro Fialho — Raymond Ros

N° 7259

Avril 2010

Domaine 1

 *R*apport
de recherche

Analysis of Adaptive Strategy Selection within Differential Evolution on the BBOB-2010 Noiseless Benchmark

Álvaro Fialho , Raymond Ros

Domaine : Mathématiques appliquées, calcul et simulation
Équipes-Projets TAO

Rapport de recherche n° 7259 — Avril 2010 — 40 pages

Abstract: This document presents the adaptive strategy selection Fitness-Based Area-Under-Curve Bandit (F-AUC-Bandit) and its use in the context of a continuous optimization algorithm: Differential Evolution. Experimental results were obtained on a testbed of noiseless functions. They demonstrate the interest of using an adaptive strategy selection technique as opposed to the naïve approach which consists in the use of a unique strategy of the optimization algorithm chosen initially.

Performance comparisons are made between F-AUC-Bandit and a uniform strategy selection approach and also other adaptive selection strategies. Finally a comparison to the state-of-the-art CMA-ES optimizer is made.

Results of the optimization algorithm using F-AUC-Bandit are still not comparable to those of CMA-ES but demonstrate a big improvement on the use of the basic DE.

Key-words: Adaptive Strategy Selection, Comparison-based, ROC Area Under Curve, Multi-Armed Bandits

Álvaro Fialho is affiliated to Microsoft Research – INRIA Joint Centre, Orsay, France.

Raymond Ros is affiliated to the TAO team-project, INRIA Saclay – Ile-de-France, Université Paris Sud, LRI, F-91405 Orsay Cedex, France.

Analyse de sélection adaptative de stratégies appliquée à l'algorithme Differential Evolution sur le banc d'essai de fonctions non-bruitées BBOB 2010

Résumé : Ce document présente l'utilisation de Fitness-based Area-Under-Curve Bandit (F-AUC-Bandit) pour faire de la sélection adaptative de stratégie dans le cadre d'un algorithme d'optimisation continue : Differential Evolution (DE). Des résultats expérimentaux sur un banc d'essai de fonctions tests sont présentés. Ils démontrent l'intérêt de l'approche adaptative comparée à l'approche naïve qui consisterait à n'utiliser qu'une unique stratégie initiale pour l'algorithme d'optimisation.

Par ailleurs, une comparaison des performances expérimentales est faite entre F-AUC-Bandit et la sélection aléatoire de stratégies de l'algorithme d'optimisation, ainsi que plusieurs techniques de sélection adaptative de stratégies. Enfin une comparaison avec l'algorithme état de l'art CMA-ES est effectuée.

Les résultats de l'algorithme d'optimisation usant de la technique F-AUC-Bandit ne sont comparables à ceux de CMA-ES mais montrent tout de même une amélioration par rapport à l'algorithme DE basique.

Mots-clés : Sélection adaptative de stratégie, aire sous la courbe ROC, bandits

Contents

1	Introduction	4
2	Differential Evolution	5
3	Presentation of <i>Adaptive Strategy Selection</i> Techniques	6
3.1	The Fitness-AUC Bandit Algorithm	6
3.2	The Fitness-SR Bandit Algorithm	7
3.3	The AUC Bandit and SR Bandit Algorithms	8
3.4	The PM-AdapSS-DE Algorithm	8
4	Experimental Settings	8
5	Results	9
5.1	<i>F-AUC-Bandit</i> versus AUC-Bandit	10
5.2	<i>F-AUC-Bandit</i> versus F-SR-Bandit	12
5.3	<i>F-AUC-Bandit</i> versus SR-Bandit	14
5.4	<i>F-AUC-Bandit</i> versus uniform DE	16
5.5	<i>F-AUC-Bandit</i> versus DE/rand/1	18
5.6	<i>F-AUC-Bandit</i> versus DE/rand/2	20
5.7	<i>F-AUC-Bandit</i> versus DE/rand-to-best/2	22
5.8	<i>F-AUC-Bandit</i> versus DE/current-to-rand/1	24
5.9	<i>F-AUC-Bandit</i> versus PM-AdapSS-DE	26
5.10	<i>F-AUC-Bandit</i> versus DMAB	28
5.11	<i>F-AUC-Bandit</i> versus AP	30
5.12	<i>F-AUC-Bandit</i> versus IPOP-CMA-ES	32
5.13	Summary Figures	34
5.14	Summary Tables	39

List of Figures

1	Sample computation of AUC reward	7
2	ECDF F-AUC versus AUC-Bandit	10
3	ECDF F-AUC versus F-SR-Bandit	12
4	ECDF F-AUC versus SR-Bandit	14
5	ECDF F-AUC versus uniform DE	16
6	ECDF F-AUC versus DE/rand/1	18
7	ECDF F-AUC versus DE/rand/2	20
8	ECDF F-AUC versus DE/rand-to-best/2	22
9	ECDF F-AUC versus DE/current-to-rand/1	24
10	ECDF F-AUC versus PM-AdapSS-DE	26
11	ECDF F-AUC versus DMAB	28
12	ECDF F-AUC versus AP	30
13	ECDF F-AUC versus IPOP-CMA-ES	32
14	ECDF F-AUC versus all DE variants	34
15	ECDF of run lengths and ratios of basic DE and <i>F-AUC-Bandit</i>	35
16	ECDF of run lengths and ratios of <i>F-AUC-Bandit</i> and <i>Adaptive Strategy Selection</i> techniques	36
17	ECDF of run lengths and ratios of <i>F-AUC-Bandit</i> and <i>Adaptive Strategy Selection</i> techniques	37

List of Tables

1	ERT Ratio table F-AUC versus AUC-Bandit	11
2	ERT Ratio table F-AUC versus F-SR-Bandit	13
3	ERT Ratio table F-AUC versus SR-Bandit	15
4	ERT Ratio table F-AUC versus uniform DE	17
5	ERT Ratio table F-AUC versus DE/rand/1	19
6	ERT Ratio table F-AUC versus DE/rand/2	21
7	ERT Ratio table F-AUC versus DE/rand-to-best/2	23
8	ERT Ratio table F-AUC versus DE/current-to-rand/1	25
9	ERT Ratio table F-AUC versus PM-AdapSS-DE	27
10	ERT Ratio table F-AUC versus DMAB	29
11	ERT Ratio table F-AUC versus AP	31
12	ERT Ratio table F-AUC versus IPOP-CMA-ES	33
13	ERT speed-up table all functions 5-D	38
14	ERT speed-up table all functions 20-D	38

1 Introduction

Differential Evolution (DE) is a very popular evolutionary algorithm. This is mainly because of its simple structure, ease of use, robustness and speed. Because of this, DE has been applied on many real-world applications, such as pattern recognition, neural network training, data mining, etc.

However, one of the features that helps making it robust with relation to so many different situations —the number of available strategies for offspring generation [15, 14]— is also responsible for adding an extra difficulty to its use, the definition of which of the available strategies should be applied to the problem at hand. Such choice is problem-dependent, and very sensitive in terms of algorithm performance, what turns to be a non-trivial decision for the user.

An off-line tuning procedure might be used to find the best strategy for the problem at hand. But, besides being computationally expensive, its result (the best single strategy) will always lead to sub-optimal behavior, as exploration tends to be more important in the beginning of the optimization process, while exploitation should be preferred when approaching to the optimum. In other words, different strategies should be applied at different moments of the optimization process, according to its “current needs” in terms of exploration and exploitation.

Defining the way such mixture of strategies will be used during the process becomes yet another optimization problem. This is the main motivation for the use of *Adaptive Strategy Selection* techniques: based on the recent performance of each strategy on the current optimization process, the strategy to be used on the generation of the next offspring is automatically chosen, while solving the problem.

A new comparison-based technique, the *F-AUC-Bandit*, has been recently proposed to this aim [5], being originally assessed in the context of adaptive operator selection within Genetic Algorithms (GAs). It uses a multi-armed bandit algorithm to select the strategy to be applied, with the Area Under the ROC Curve paradigm [4] being used to assess the performance of each strategy, based on the ranks of the fitnesses of the generated offspring, what makes it

totally invariant with relation to monotonous transformations over the fitness function.

In this report we provide an empirical validation of *Adaptive Strategy Selection* coupling it with a DE algorithm, and analyzing it in the context of continuous optimization, on the BBOB-2010 noiseless benchmarking suite.

The DE algorithm coupled with *F-AUC-Bandit* is compared to the basic DE using one single initial strategy. Also, as for strategy selection techniques, we compare *F-AUC-Bandit* to:

- an uniform selection between the strategies in DE,
- the *PM-AdapSS-DE* [8] adaptive strategy selection technique,
- a method which uses the fitness-based sum of ranks [5] as credit assignment scheme (*F-SR-Bandit*),
- and the counterparts of *F-AUC-Bandit* and *F-SR-Bandit*, using fitness improvements instead of the ranking using fitness values.

Furthermore some comparisons with the state-of-the-art CMA-ES optimizer are made [2].

As preliminaries, the DE algorithm will be briefly overviewed in Section 2. The adaptive strategy selection techniques are briefly described in Section 3 (we refer the reader to the original papers [5, 8] for a more complete view). Section 4 presents the experimental settings that were used to generate the results presented in Section 5.

2 Differential Evolution

Differential Evolution (DE) [15, 14] is an evolutionary algorithm that uses a differential mutation procedure that consists in the addition of the weighted difference of two population vectors to a third vector. DE loops over the following steps:

Mutation Many different mutation schemes or *strategies* can be considered. We list a few here:

- 1) “DE/rand/1”: $\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3})$
- 2) “DE/rand/2”: $\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) + F \cdot (\mathbf{x}_{r_4} - \mathbf{x}_{r_5})$
- 3) “DE/rand-to-best/2”: $\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{best} - \mathbf{x}_{r_1}) + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) + F \cdot (\mathbf{x}_{r_4} - \mathbf{x}_{r_5})$
- 4) “DE/current-to-rand/1”:
 $\mathbf{v}_i = \mathbf{x}_i + F \cdot (\mathbf{x}_{r_1} - \mathbf{x}_i) + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3})$

where i is in $1, \dots, NP$ with NP being the population size, \mathbf{x}_i represents the current individual, \mathbf{x}_{best} is the best individual in the current generation, r_1, r_2, r_3, r_4, r_5 are individuals randomly chosen from the population, being $r_1 \neq r_2 \neq r_3 \neq r_4 \neq r_5 \neq i$. F is a parameter, the mutation scaling factor, in the range $]0, 2]$,

Crossover \vec{u}_i is the resulting individual of the crossover between the parent \vec{x}_i and the mutant candidate \vec{v}_i which is generated by choosing component by component between those of \vec{v}_i and \vec{x}_i with probability CR and $1 - CR$ respectively (CR being a parameter), with the exception that one random component of \vec{u}_i must correspond to that of \vec{v}_i ,

Selection the candidate \vec{u}_i replaces \vec{x}_i if there is an improvement.

The parameters CR , F and the choice of the strategies is a sensitive issue in the sense that choosing the right strategy and setting the algorithm parameters is very much dependent on the type of problem considered. This especially motivates the use of adaptive strategy selection techniques.

3 Presentation of Adaptive Strategy Selection Techniques

To do *Adaptive Strategy* (or Operator) *Selection*, there is the need of defining how the impact of the application of a given strategy is assessed, *i.e.*, how to reward the strategy after its “production”, which is referred to as the *Credit Assignment* mechanism; and based on these assessments, there is the need of defining how to select the next strategy to be applied, which is called the *Strategy Selection* scheme. The techniques being analyzed in this report will be briefly described in the following, focusing on how they handle these two issues.

3.1 The Fitness-AUC Bandit Algorithm

The *F-AUC-Bandit* algorithm, recently proposed in the context of GAs [5], uses the Area Under the ROC Curve (AUC) paradigm to assess the empirical quality of each strategy. The AUC is a criterion originally used in *Machine Learning* to compare binary classification rules [4]. In the context of *Adaptive Strategy Selection*, it shows how good one strategy is, by comparing the rewards received after its recent applications with the rewards received by the others.

Instead of being calculated based on the received raw rewards, in this work the AUC uses the ranks of these rewards, what improves its robustness with relation to different problems (there is no need of re-scaling the algorithm to the different possible ranges of rewards). Besides, by directly using the rank of the fitnesses of the generated offspring, instead of the commonly used fitness improvements, it becomes a total comparison-based method, invariant with relation to monotonous transformations over the fitness function. This is what we refer to as the *Fitness-based AUC* credit assignment scheme. In case the generated offspring does not improve over its parent, a null reward is assigned.

Figure 1, reproduced from [5], illustrates an example computation of the AUC. Briefly, it is the total area upper bounded by the *Receiving Operator Curve* (ROC), represented by the solid line in the example. Computing the quality of a given strategy consists of going down the sorted list of raw rewards, and drawing, starting from the origin, a vertical segment each time the strategy under assessment is found in the list, a horizontal one otherwise, and a diagonal in case of ties.

In this example, for the sake of clarity, each rank position has the same weight on the calculation of the reward, *i.e.*, each segment has the same length

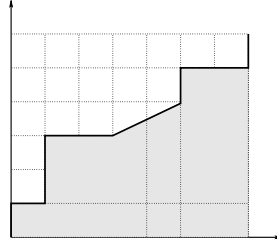


Figure 1: Sample computation of AUC reward: only two operators are involved, and the sorted list contains the operators in the order (1 2 1 1 2 2 [2 2 1] 1 2 2 1), with [2 2 1] meaning that these 3 positions have the same raw reward, leading to the diagonal line between points (3 3) and (5 4) (dotted lines are spaced by 1). In case of decay, the width of the squares would decrease leftward and upward.

than the others, no matter its ranking. But it is clear that the initial rank positions (the best raw rewards) should have a higher impact on this quality estimation. To this aim, a decay factor can be applied. Being W the size of the sliding window that stores the recent raw rewards received by all the strategies, and r the rank position of a given reward, the length of its segment in the ROC curve (*i.e.*, its importance in the AUC computation) can be calculated as $D^r(W - r)$, with $D \in]0, 1]$ being the decay factor that defines how skewed is this ranking distribution. A linear decay is achieved by using $D = 1$; smaller D , faster the decay.

A multi-armed bandit technique, based on the UCB multi-armed bandit formula [1, 5], is then used to select the next strategy to be applied, according to the presented quality estimation. The main difference is that, as there is no much sense in calculating statistics over statistics, in this case the empirical quality used by the UCB formula is equal to the last received reward (the AUC computation), instead of being an average of the received rewards, since the AUC efficiently summarizes the up-to-date quality of the strategy with relation to the others.

Besides the decay factor D , the *F-AUC-Bandit* algorithm requires the definition of a scaling factor C , that is used to balance the importance of its exploration and exploitation terms; and also the credits sliding window size W .

3.2 The Fitness-SR Bandit Algorithm

The **Sum of Ranks** (SR) method, as its name already says, credits the operators with the sum of the ranks of the rewards given after its applications, normalized by the sum of all the rank-values, so that the sum of the credits assigned to all operators sum up to 1. Being K the number of operators, the operator i is rewarded at time t as follows:

$$SR_{i,t} = \frac{\sum_{op_r=i} D^r(W - r)}{\sum_{r=1}^W D^r(W - r)} \quad (1)$$

3.3 The AUC Bandit and SR Bandit Algorithms

These techniques are counterparts of the *F-AUC-Bandit* and the *F-SR-Bandit* that uses fitness improvements instead of the fitness values for ranking the strategies.

3.4 The PM-AdapSS-DE Algorithm

The *PM-AdapSS-DE* algorithm is much simpler, although presenting competitive results, as shown in its original paper [8]. The credit assignment scheme awards the strategy with the absolute average of the raw rewards recently received by it. The raw reward in this case is the relative fitness improvement (the improvement achieved by the offspring over its parent, normalized by the fitness of the best-so-far individual).

The Probability Matching (PM) technique [7] uses this received credit to update the empirical quality estimate it keeps for each strategy, with the weight of the received reward being ruled by a user-defined parameter, the adaptation rate $\alpha \in]0, 1]$. The probability of selection of each strategy is then updated proportionally to its empirical estimate with relation to the others. A minimal probability $p_{min} \in [0, 1]$ might be applied, so that no operator gets “lost” during the process [16]. With this, every time a strategy needs to be applied, it is selected from the set of available ones by a roulette-wheel-like process over these up-to-date probabilities.

4 Experimental Settings

We test variants of DE on some continuous optimization problems given by the BBOB 2010 experimental framework [10]. BBOB 2010 provides with a whole experimental set-up for testing continuous optimizers. In particular, BBOB 2010 provides test problems [6, 11] in dimension from two to forty in the form of function instances: each of the twenty-four functions of the noiseless testbed has fifteen instances, totalizing 720 problems for each dimension.

Each of the mutation schemes presented in Section 2 are tested individually. We also test a variant of DE with uniform selection of the strategy during the optimization process. Furthermore we consider *PM-AdapSS-DE* and *F-AUC-Bandit* applied to DE.

For the parameters of DE, the population size NP was fixed at 10 times the dimension of the search space, and the mutation scaling factor F was set to 0.5. Although a crossover rate CR of 0.9 is usually advocated, it was chosen to set it to 1.0 for this benchmarking exercise, as in this way the DE becomes invariant with relation to rotation, and entirely dependent on the application of the mutation strategies [12].

The off-line tuning of the adaptive schemes *PM-AdapSS-DE*, *F-AUC-Bandit*, *AUC-Bandit*, *F-SR-Bandit* and *SR-Bandit* was done using the F-Race technique [3] as in [5]. The F-Race technique starts to eliminate the configurations after one run over all instances of a given dimension, with the Friedman’s two-way analysis of variances by ranks statistical test being applied at a rate of 95%, up to 10 runs or just one configuration left. For the *F-AUC-Bandit* and the *AUC-Bandit*, the following parameter values were tried: scaling factor $C \in \{.1, .5, 1, 5, 10\}$, decay factor $D \in \{.5, 1\}$, and window size $W \in \{50, 100, 500\}$,

summing up to thirty configurations. And for the *PM-AdapSS-DE*, the minimal probability $p_{min} \in \{0, .05, .1, .2\}$, and the adaptation rate $\alpha \in \{.1, .3, .6, .9\}$ were tried, totalizing sixteen configurations.

Although different tuning procedures were executed independently for each dimension $\in \{5, 20\}$ (a kind of representative set of all the analyzed dimensions), the same configuration was found to be the best for both dimensions, for both techniques. This confirms the already mentioned robustness gain provided by the use of relative rewards instead of raw ones: the parameters of the adaptive scheme do not need to be re-scaled for each problem, a same parameter configuration might be well-performing on an entire class of problems. In the case of the *F-AUC-Bandit*, given its comparison-based property, this is ensured to be true for all the class of fitness functions defined by monotonous transformations over the original one. After this tuning phase, the *F-AUC-Bandit* was set up with the following parameter values: $\{D = .5, C = .5, W = 50\}$, and the *PM-AdapSS-DE* was assessed with $\{p_{min} = 0, \alpha = .6\}$.

The experiments were performed following the BBOB guidelines [10], with the maximum number of evaluations being fixed at 10^5 times the dimension. The aforementioned parameter values were used on all the experiments, for all dimensions, thus the crafting effort for techniques is equal to zero.

Finally we also consider another algorithm to compare with the variants of DE we presented. The state-of-the-art CMA-ES optimizer with an Increasing POPulation (IPOP-CMA-ES) size restart strategy [2] was tested on the BBOB 2010 test suite with the exact same parameter tuning than that used in [9].

5 Results

Results from experiments according to [10] are presented in this section. The **expected running time (ERT)**, used in the figures and table, depends on a given target function value, $f_t = f_{opt} + \Delta f$, 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 [10, 13]. **Statistical significance** is tested with the rank-sum test for a given target Δf_t 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.

All algorithms are compared to *F-AUC-Bandit*.

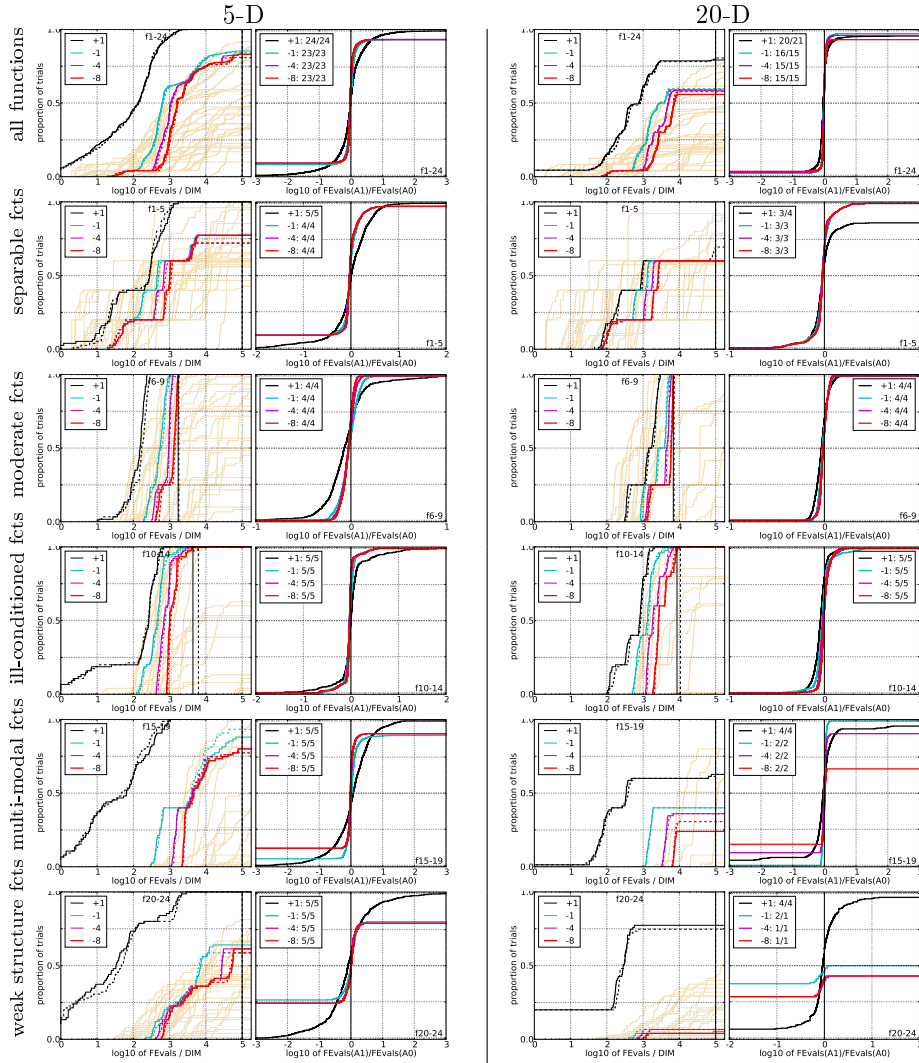
5.1 *F-AUC-Bandit* versus AUC-Bandit

Figure 2: 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 F-AUC (solid) and AUC-Bandit (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 F-AUC divided by AUC-Bandit, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f₁	11	12	12	12	12	15/15	f₁	43	43	43	43	43	15/15
0:auc	8	71	144	209	281	15/15	0:auc	100	275	449	625	799	15/15
1:f-auc	6	67	132	203	266	15/15	1:f-auc	93	265	431	597	763	15/15
f₂	83	88	90	92	94	15/15	f₂	385	387	390	391	393	15/15
0:auc	20	28	38	46	54	15/15	0:auc	51	72	91	110	129	15/15
1:f-auc	18	26*	35*	42*	50*	15/15	1:f-auc	48	68	86	105	123	15/15
f₃	716	1637	1646	1650	1654	15/15	f₃	5066	7635	7643	7646	7651	15/15
0:auc	3	215	214	214	214	9/15	0:auc	789	∞	∞	∞	∞	2.0e6
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	2.0e6
f₄	809	1688	1817	1886	1903	15/15	f₄	4722	7666	7700	7758	1.4e5	9/15
0:auc	4	∞	∞	∞	∞	5.0e5	0/auc	∞	∞	∞	∞	∞	2.0e6
1:f-auc	6	∞	∞	∞	∞	5.0e5	0/auc	∞	∞	∞	∞	∞	2.0e6
f₅	10	10	10	10	10	15/15	f₅	41	41	41	41	41	15/15
0:auc	12	20	20	20	20	15/15	0:auc	47	59	59	59	59	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42	53	54	54	54	15/15
f₆	114	281	580	1038	1332	15/15	f₆	1296	3413	5220	6728	8409	15/15
0:auc	8	8	7	5	6	15/15	0:auc	21	15	15	15	15	15/15
1:f-auc	6	8	7	5	5	15/15	1:f-auc	19	14*3	14*3	14*3	14*3	15/15
f₇	24	1171	1572	1572	1597	15/15	f₇	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:auc	14	1	1	1	2	15/15	0:auc	6	2	2	2	2	15/15
1:f-auc	13	1	1*	1*	1	15/15	1:f-auc	5*	2	2	2	2	15/15
f₈	73	336	391	410	422	15/15	f₈	2039	4040	4219	4371	4484	15/15
0:auc	15	11	12	14	15	15/15	0:auc	24	22	24	25	26	15/15
1:f-auc	13	11	11	13	14	15/15	1:f-auc	23	23	24	25	26	15/15
f₉	35	214	300	335	369	15/15	f₉	1716	3277	3455	3594	3727	15/15
0:auc	31	19	17	17	18	15/15	0:auc	26	26	28	29	30	15/15
1:f-auc	24	16	16	16	17	15/15	1:f-auc	26	27	29	30	31	15/15
f₁₀	349	574	626	829	880	15/15	f₁₀	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:auc	5	4	5	5	6	15/15	0:auc	3	3	2	3	3	15/15
1:f-auc	4	4	5	5	5	15/15	1:f-auc	2	2*	2*	2	3	15/15
f₁₁	143	763	1177	1467	1673	15/15	f₁₁	1002	6278	9762	1.2e4	1.5e4	15/15
0:auc	6	2	2	2	3	15/15	0:auc	8	3	2	3	3	15/15
1:f-auc	6	2	2*	2*	2*	15/15	1:f-auc	7*	2	2	2	2	15/15
f₁₂	108	371	461	1303	1494	15/15	f₁₂	1042	2740	4140	1.2e4	1.4e4	15/15
0:auc	24	15	16	7	8	15/15	0:auc	27	18	19	9	10	15/15
1:f-auc	22	13	14	6	7	15/15	1:f-auc	27	19	20	9	10	15/15
f₁₃	132	250	1310	1752	2255	15/15	f₁₃	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:auc	10	12	3	3	3	15/15	0:auc	26	12	2	3	3	15/15
1:f-auc	10	11	3*	3*2	3*	15/15	1:f-auc	25	11	2	2	3	15/15
f₁₄	10	58	139	251	476	15/15	f₁₄	75	304	932	1648	1.6e4	15/15
0:auc	2	15	15	12	9	15/15	0:auc	42	40	23	19	3	15/15
1:f-auc	2	15	13	11	8	15/15	1:f-auc	33	38	23	19	3	15/15
f₁₅	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f₁₅	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:auc	4	1	1	1	1	15/15	0:auc	481	∞	∞	∞	∞	2.0e6
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	2.0e6
f₁₆	120	2662	1.0e4	1.2e4	1.2e4	15/15	f₁₆	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:auc	4	15	21	19	19	11/15	0:auc	∞	∞	∞	∞	∞	2.0e6
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	2.0e6
f₁₇	5	899	3669	6351	7934	15/15	f₁₇	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:auc	4	2	1	1	1	15/15	0:auc	25	7	7	4	5	12/15
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23	7	2	2	5	13/15
f₁₈	103	3968	9280	1.1e4	1.2e4	15/15	f₁₈	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:auc	4	0.7	0.7	0.9	1	15/15	0:auc	12	2	3	5	6	11/15
1:f-auc	4	0.8	0.7	0.9	1.0	15/15	1:f-auc	11	2	3	11	28	5/15
f₁₉	1	242	1.2e5	1.2e5	1.2e5	15/15	f₁₉	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:auc	35	1498	29	28	28	2/15	0:auc	1379	∞	∞	∞	∞	2.0e6
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315	∞	∞	∞	∞	2.0e6
f₂₀	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f₂₀	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:auc	14	9	6	6	6	9/15	0:auc	48	∞	∞	∞	∞	2.0e6
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40*	∞	∞	∞	∞	2.0e6
f₂₁	41	1674	1705	1729	1757	14/15	f₂₁	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:auc	3	151	149	147	145	10/15	0:auc	266	285	274	258	229	5/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12	568	547	515	456	3/15
f₂₂	71	938	1008	1040	1068	14/15	f₂₂	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:auc	6	469	437	424	414	8/15	0:auc	1086	∞	∞	∞	∞	2.0e6
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	2.0e6
f₂₃	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f₂₃	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:auc	2	2	3	5	7	15/15	0:auc	1	∞	∞	∞	∞	2.0e6
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440	∞	∞	∞	2.0e6
f₂₄	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f₂₄	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:auc	5	0.5	0.3	0.3	0.3	2/15	0:auc	∞	∞	∞	∞	∞	2.0e6
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	2.0e6

Table 1: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:auc is AUC-Bandit and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

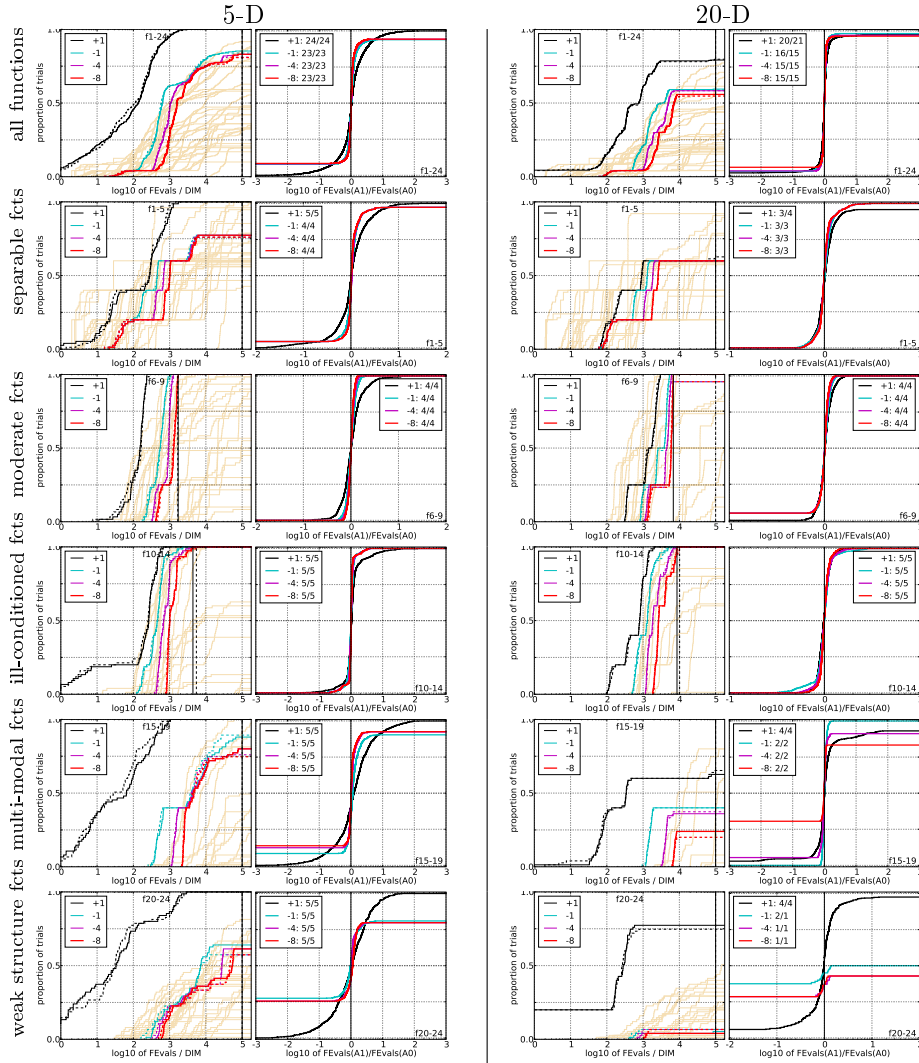
5.2 *F-AUC-Bandit* versus F-SR-Bandit

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 F-AUC (solid) and F-SR-Bandit (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 F-AUC divided by F-SR-Bandit, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f₁	11	12	12	12	12	15/15	f₁	43	43	43	43	43	15/15
0:f-sr	6	69	129	197	262	15/15	0:f-sr	90	259	430	604	766	15/15
1:f-auc	6	67	132	203	266	15/15	1:f-auc	93	265	431	597	763	15/15
f₂	83	88	90	92	94	15/15	f₂	385	387	390	391	393	15/15
0:f-sr	18	25	33	42	49	15/15	0:f-sr	48	67	86	106	123	15/15
1:f-auc	18	26	35	42	50	15/15	1:f-auc	48	68	86	105	123	15/15
f₃	716	1637	1646	1650	1654	15/15	f₃	5066	7635	7643	7646	7651	15/15
0:f-sr	4	87	88	88	88	12/15	0:f-sr	2872	∞	∞	∞	∞	0/15
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f₄	809	1688	1817	1886	1903	15/15	f₄	4722	7666	7700	7758	1.4e5	9/15
0:f-sr	4	∞	∞	∞	∞	0/15	0:f-sr	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f₅	10	10	10	10	10	15/15	f₅	41	41	41	41	41	15/15
0:f-sr	12	20	20	20	20	15/15	0:f-sr	41	51	51	51	51	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42	53	54	54	54	15/15
f₆	114	281	580	1038	1332	15/15	f₆	1296	3413	5220	6728	8409	15/15
0:f-sr	7	9	7	5	5	15/15	0:f-sr	18	13	13	14	14	15/15
1:f-auc	6	8	7	5	5	15/15	1:f-auc	19	14	14	14	14	15/15
f₇	24	1171	1572	1572	1597	15/15	f₇	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:f-sr	10	1	1	1	1	15/15	0:f-sr	5	17	10	10	10	14/15
1:f-auc	13	1	1	1	1	15/15	1:f-auc	5	2	2	2	2	15/15
f₈	73	336	391	410	422	15/15	f₈	2039	4040	4219	4371	4484	15/15
0:f-sr	13	10	11	13	14	15/15	0:f-sr	22	97	96	94	94	13/15
1:f-auc	13	11	11	13	14	15/15	1:f-auc	23	23	24	25	26	15/15
f₉	35	214	300	335	369	15/15	f₉	1716	3277	3455	3594	3727	15/15
0:f-sr	25	17	15	16	17	15/15	0:f-sr	26	27	29	30	31	15/15
1:f-auc	24	16	16	16	17	15/15	1:f-auc	26	27	29	30	31	15/15
f₁₀	349	574	626	829	880	15/15	f₁₀	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:f-sr	4	4	5	5	5	15/15	0:f-sr	2	2	2	2	3	15/15
1:f-auc	4	4	5	5	5	15/15	1:f-auc	2	2	2	2	3	15/15
f₁₁	143	763	1177	1467	1673	15/15	f₁₁	1002	6278	9762	1.2e4	1.5e4	15/15
0:f-sr	5	2	2	2	2	15/15	0:f-sr	7	2	2	2	3	15/15
1:f-auc	6	2	2	2	2	15/15	1:f-auc	7	2	2	2	2	15/15
f₁₂	108	371	461	1303	1494	15/15	f₁₂	1042	2740	4140	1.2e4	1.4e4	15/15
0:f-sr	23	14	15	7	7	15/15	0:f-sr	27	21	21	10	10	15/15
1:f-auc	22	13	14	6	7	15/15	1:f-auc	27	19	20	9	10	15/15
f₁₃	132	250	1310	1752	2255	15/15	f₁₃	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:f-sr	10	11	3	3	3	15/15	0:f-sr	25	11	2	2	2	15/15
1:f-auc	10	11	3	3	3	15/15	1:f-auc	25	11	2	2	3	15/15
f₁₄	10	58	139	251	476	15/15	f₁₄	75	304	932	1648	1.6e4	15/15
0:f-sr	1	15	14	11	8	15/15	0:f-sr	34	37	22	19	3	15/15
1:f-auc	2	15	13	11	8	15/15	1:f-auc	33	38	23	19	3	15/15
f₁₅	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f₁₅	3.0e4	3.1e5	3.2e5	3.5e5	4.6e5	15/15
0:f-sr	4	7	7	7	7	12/15	0:f-sr	229	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	0/15
f₁₆	120	2662	1.0e4	1.2e4	1.2e4	15/15	f₁₆	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:f-sr	3	25	7	10	9	13/15	0:f-sr	∞	∞	∞	∞	∞	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f₁₇	5	899	3669	6351	7934	15/15	f₁₇	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:f-sr	5	2	1	1	1	15/15	0:f-sr	19	6	2	4	8	12/15
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23	7	2	2	5	13/15
f₁₈	103	3968	9280	1.1e4	1.2e4	15/15	f₁₈	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:f-sr	4	0.7	0.7	0.9	1.0	15/15	0:f-sr	11	2	3	9	56	3/15
1:f-auc	4	0.8	0.7	0.9	1.0	15/15	1:f-auc	11	2	3	11	28	5/15
f₁₉	1	242	1.2e5	1.2e5	1.2e5	15/15	f₁₉	1	3.4e5	6.2e6	7.6e6	6.7e6	15/15
0:f-sr	37	1120	60	59	59	1/15	0:f-sr	1347	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315	∞	∞	∞	∞	0/15
f₂₀	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f₂₀	82	3.1e6	5.6e6	5.6e6	5.6e6	14/15
0:f-sr	9	15	11	11	11	7/15	0:f-sr	42	∞	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40	∞	∞	∞	∞	0/15
f₂₁	41	1674	1705	1729	1757	14/15	f₂₁	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:f-sr	4	110	108	107	106	11/15	0:f-sr	11	284	274	258	229	5/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12	568	547	515	456	3/15
f₂₂	71	938	1008	1040	1068	14/15	f₂₂	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:f-sr	4	612	569	552	539	7/15	0:f-sr	1574	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f₂₃	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f₂₃	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:f-sr	1	2	3	5	6*	15/15	0:f-sr	2	∞	∞	∞	∞	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440	∞	∞	∞	0/15
f₂₄	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f₂₄	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:f-sr	5	0.3	0.2	0.2	0.2	3/15	0:f-sr	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 2: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:f-sr is F-SR-Bandit and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

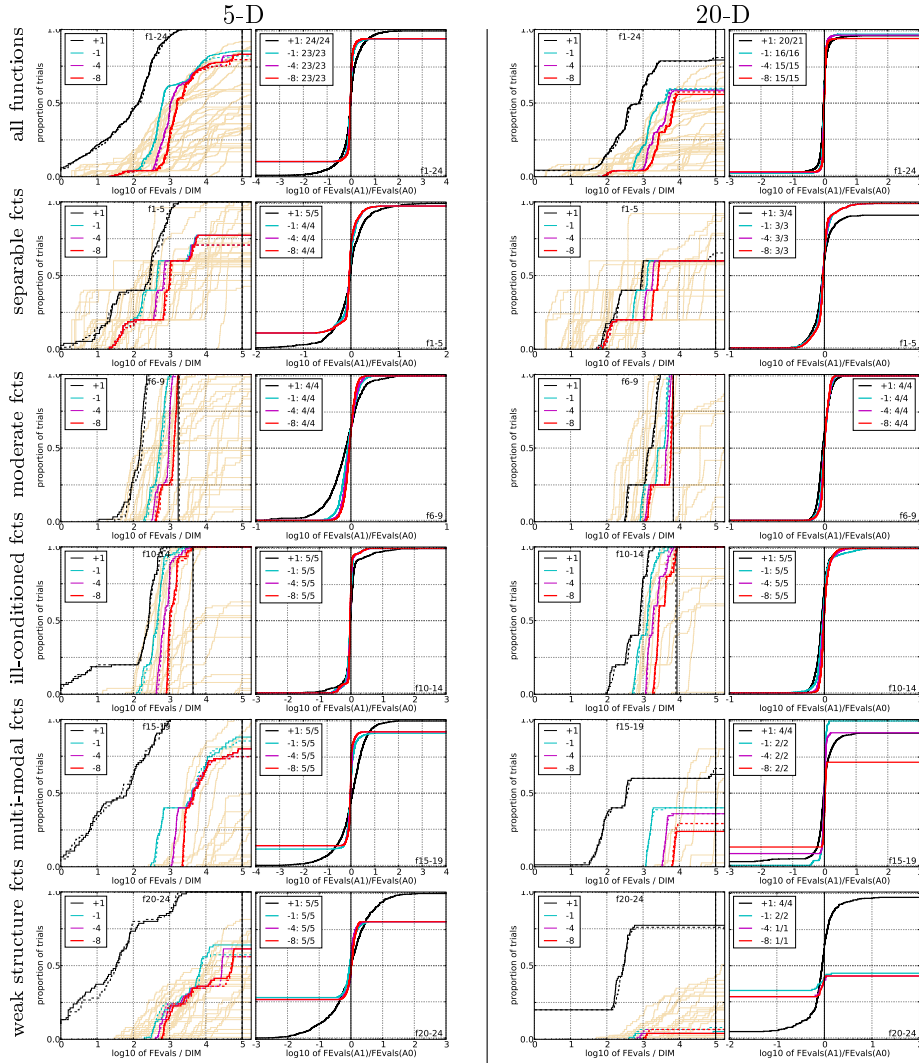
5.3 *F-AUC-Bandit* versus SR-Bandit

Figure 4: 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 F-AUC (solid) and SR-Bandit (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 F-AUC divided by SR-Bandit, 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 (F-AUC first).

5-D							20-D							
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15	
0:sr	5	74	139	213	282	15/15	0:sr	99	269	436	607	778	15/15	
1:f-auc	6	6	67	132	203	266	15/15	1:f-auc	93	265	431	597	763	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15	
0:sr	19	27	37	46	53	15/15	0:sr	50	70	89	108	126	15/15	
1:f-auc	18	26	35	42*	50	15/15	1:f-auc	48	68	86	105	123	15/15	
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15	
0:sr	4	278	277	277	277	8/15	0:sr	1372	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	2.0e6	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15	
0:sr	6	∞	∞	∞	∞	5.0e5	0:sr	∞	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	6	∞	∞	∞	∞	5.0e5	1:f-auc	∞	∞	∞	∞	∞	2.0e6	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15	
0:sr	17	28	28	28	28	15/15	0:sr	45	56	56	56	56	15/15	
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42	53	54	54	54	15/15	
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15	
0:sr	8	9	7	6	6	15/15	0:sr	20	14	14	14	14	15/15	
1:f-auc	6	8	7	5	5	15/15	1:f-auc	19	14	14	14	14	15/15	
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15	
0:sr	15	1	1	1	1	15/15	0:sr	6	2	2	2	2	15/15	
1:f-auc	13	1	1	1	1	15/15	1:f-auc	5	2	2	2	2	15/15	
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15	
0:sr	14	11	12	13	15	15/15	0:sr	22	21	23	24	25	15/15	
1:f-auc	13	11	11	13	14	15/15	1:f-auc	23	23	24	25	26	15/15	
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15	
0:sr	29	17	15	16	17	15/15	0:sr	26	26	28	29	30	15/15	
1:f-auc	24	16	16	16	17	15/15	1:f-auc	26	27	29	30	31	15/15	
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15	
0:sr	5	4	5	5	6	15/15	0:sr	3	3	2	3	3	15/15	
1:f-auc	4	4	5	5	5	15/15	1:f-auc	2	2	2	2	3	15/15	
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15	
0:sr	6	2	2	2	3	15/15	0:sr	8	3	2	2	3	15/15	
1:f-auc	6	2	2	2	2	15/15	1:f-auc	7	2	2	2	2	15/15	
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15	
0:sr	26	15	16	7	7	15/15	0:sr	27	17	18	9	9	15/15	
1:f-auc	22	13	14	6	7	15/15	1:f-auc	27	19	20	9	10	15/15	
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15	
0:sr	10	11	3	3	3	15/15	0:sr	26	11	2	2	2	15/15	
1:f-auc	10	11	3	3	3	15/15	1:f-auc	25	11	2	2	3	15/15	
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15	
0:sr	2	17	15	12	9	15/15	0:sr	37	40	23	20	3	15/15	
1:f-auc	2	15	13	11	8*	15/15	1:f-auc	33	38	23	19	3	15/15	
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15	
0:sr	5	5	5	5	5	13/15	0:sr	185	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	2.0e6	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15	
0:sr	8	45	17	15	14	12/15	0:sr	∞	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	2.0e6	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15	
0:sr	5	2	1	1	1	15/15	0:sr	21	7	2	2	3	14/15	
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23	7	2	2	5	13/15	
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15	
0:sr	3	0.7	0.7	0.9	1	15/15	0:sr	12	2	8	11	13	8/15	
1:f-auc	4	0.8	0.7	0.9	1.0	15/15	1:f-auc	11	2	3	11	28	5/15	
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	2.6e6	7e6	6.7e6	15/15	
0:sr	31	2488	60	60	59	1/15	0:sr	1472	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315	∞	∞	∞	∞	2.0e6	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15	
0:sr	10	12	8	8	8	8/15	0:sr	48	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40*	∞	∞	∞	∞	2.0e6	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15	
0:sr	5	200	197	195	192	9/15	0:sr	12	285	274	258	229	5/15	
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12	568	547	515	456	3/15	
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15	
0:sr	4	358	334	324	316	9/15	0:sr	1086	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	2.0e6	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15	
0:sr	2	2	3	5	6	15/15	0:sr	2	433	∞	∞	∞	2.0e6	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440	∞	∞	∞	2.0e6	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15	
0:sr	4	0.5	0.7	0.5	0.5	1/15	0:sr	∞	∞	∞	∞	∞	2.0e6	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	2.0e6	0/15

Table 3: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 – f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:sr is SR-Bandit and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

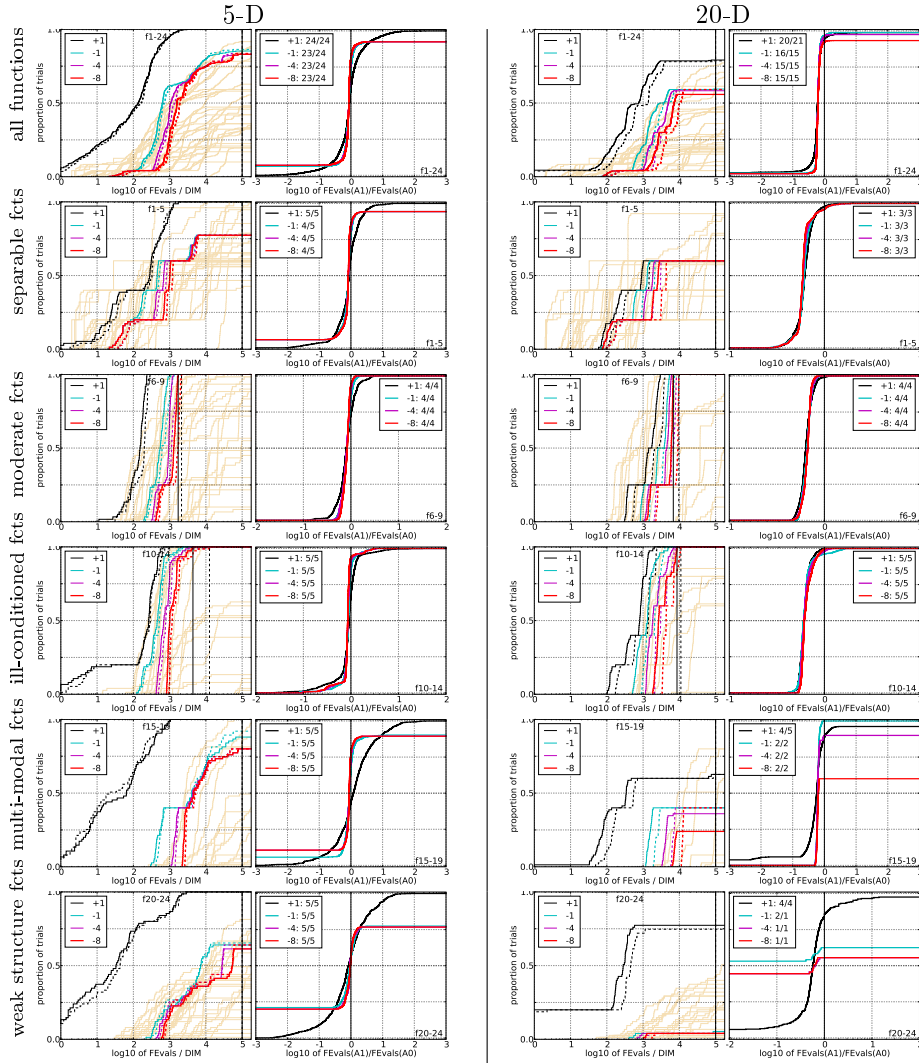
5.4 *F-AUC-Bandit* versus uniform DE

Figure 5: 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 F-AUC (solid) and uniform DE (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 F-AUC divided by uniform DE, 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 (F-AUC first).

5-D								20-D							
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ		Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	
f_1	11	12	12	12	12	15/15		f_1	43	43	43	43	43	15/15	
0:unif	9	80	155	236	313	15/15		0:unif	146	442	734	1027	1319	15/15	
1:f-auc	6	67	132*2	203*2	266*3	15/15		1:f-auc	93*3	265*3	431*3	597*3	763*3	15/15	
f_2	83	88	90	92	94	15/15		f_2	385	387	390	391	393	15/15	
0:unif	20	30	40	50	58	15/15		0:unif	77	108	140	172	203	15/15	
1:f-auc	18*	26*3	35*3	42*3	50*3	15/15		1:f-auc	48*3	68*3	86*3	105*3	123*3	15/15	
f_3	716	1637	1646	1650	1654	15/15		f_3	5066	7635	7643	7646	7651	15/15	
0:unif	3	125	125	125	126	11/15		0:unif	∞	∞	∞	∞	∞	0/15	
1:f-auc	3	59	60	60	60	13/15		1:f-auc	∞	∞	∞	∞	∞	0/15	
f_4	809	1688	1817	1886	1903	15/15		f_4	4722	7666	7700	7758	1.4e5	9/15	
0:unif	5	1941	1803	1739	1723	2/15		0:unif	∞	∞	∞	∞	∞	0/15	
1:f-auc	6	∞	∞	∞	∞	0/15		1:f-auc	∞	∞	∞	∞	∞	0/15	
f_5	10	10	10	10	10	15/15		f_5	41	41	41	41	41	15/15	
0:unif	20	35	35	35	35	15/15		0:unif	69	85	86	86	86	15/15	
1:f-auc	15	24	24	24	24	15/15		1:f-auc	42*2	53*	54*	54*	54*	15/15	
f_6	114	281	580	1038	1332	15/15		f_6	1296	3413	5220	6728	8409	15/15	
0:unif	9	10	8	6	6	15/15		0:unif	29	20	20	20	20	15/15	
1:f-auc	6*	8*2	7*	5*3	5*3	15/15		1:f-auc	19*3	14*3	14*3	14*3	14*3	15/15	
f_7	24	1171	1572	1572	1597	15/15		f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15	
0:unif	14	1	1	1	2	15/15		0:unif	8	3	3	3	3	15/15	
1:f-auc	13	1	1	1	1*2	15/15		1:f-auc	5*3	2*3	2*3	2*3	2*3	15/15	
f_8	73	336	391	410	422	15/15		f_8	2039	4040	4219	4371	4484	15/15	
0:unif	15	14	17	18	20	15/15		0:unif	34	33	35	37	39	15/15	
1:f-auc	13	11	11*2	13*2	14*2	15/15		1:f-auc	23*3	23*3	24*3	25*3	26*3	15/15	
f_9	35	214	300	335	369	15/15		f_9	1716	3277	3455	3594	3727	15/15	
0:unif	30	21	21	22	22	15/15		0:unif	41	41	43	45	46	15/15	
1:f-auc	24	16*	16*2	16*2	17*2	15/15		1:f-auc	26*3	27*3	29*3	30*3	31*3	15/15	
f_{10}	349	574	626	829	880	15/15		f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15	
0:unif	4	5	6	5	6	15/15		0:unif	4	4	4	4	5	15/15	
1:f-auc	4	4*	5*	5*2	5*3	15/15		1:f-auc	2*3	2*3	2*3	2*3	3*3	15/15	
f_{11}	143	763	1177	1467	1673	15/15		f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15	
0:unif	6	2	2	3	3	15/15		0:unif	11	4	4	4	4	15/15	
1:f-auc	6	2*2	2*3	2*3	2*3	15/15		1:f-auc	7*3	2*3	2*3	2*3	2*3	15/15	
f_{12}	108	371	461	1303	1494	15/15		f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15	
0:unif	28	21	22	10	11	15/15		0:unif	44	25	26	12	13	15/15	
1:f-auc	22*2	13	14	6	7	15/15		1:f-auc	27*3	19	20*	9*2	10*3	15/15	
f_{13}	132	250	1310	1752	2255	15/15		f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15	
0:unif	11	13	4	4	4	15/15		0:unif	43	19	4	4	4	15/15	
1:f-auc	10*	11*2	3*3	3*3	3*3	15/15		1:f-auc	25*3	11*3	2*3	2*3	3*3	15/15	
f_{14}	10	58	139	251	476	15/15		f_{14}	75	304	932	1648	1.6e4	15/15	
0:unif	2	16	15	13	9	15/15		0:unif	53	64	38	31	4	15/15	
1:f-auc	2	15	13*	11*3	8*3	15/15		1:f-auc	33*3	38*3	23*3	19*3	3*3	15/15	
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15		f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15	
0:unif	5	3	3	3	3	14/15		0:unif	979	∞	∞	∞	∞	0/15	
1:f-auc	5	7	7	7	7	12/15		1:f-auc	474	∞	∞	∞	∞	0/15	
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15		f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15	
0:unif	4	20	17	16	15	12/15		0:unif	2.1e4	∞	∞	∞	∞	0/15	
1:f-auc	6	31	12	11	10	13/15		1:f-auc	∞	∞	∞	∞	∞	0/15	
f_{17}	5	899	3669	6351	7934	15/15		f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15	
0:unif	4	3	2	1	2	15/15		0:unif	38	10	3	3	3	15/15	
1:f-auc	6	2*2	1*3	1*3	1*3	15/15		1:f-auc	23	7*3	2*3	2*3	5	13/15	
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15		f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15	
0:unif	4	0.8	0.7	0.9	1	15/15		0:unif	18	3	2	1	2	15/15	
1:f-auc	4	0.8	0.7	0.9	1.0*2	15/15		1:f-auc	11*3	2*3	3	11	28	5/15	
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15		f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15	
0:unif	35	1630	13	13	13	4/15		0:unif	2846	∞	∞	∞	∞	0/15	
1:f-auc	29	1726	11	11	10	5/15		1:f-auc	1315*2	∞	∞	∞	∞	0/15	
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15		f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15	
0:unif	11	9	6	6	6	9/15		0:unif	76	∞	∞	∞	∞	0/15	
1:f-auc	7	7	5	5	5	10/15		1:f-auc	40*3	∞	∞	∞	∞	0/15	
f_{21}	41	1674	1705	1729	1757	14/15		f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15	
0:unif	4	76	75	74	74	12/15		0:unif	21	568	548	516	457	3/15	
1:f-auc	5	110	109	107	106	11/15		1:f-auc	12*3	568	547	515	456	3/15	
f_{22}	71	938	1008	1040	1068	14/15		f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15	
0:unif	7	469	437	424	414	8/15		0:unif	1580	∞	∞	∞	∞	0/15	
1:f-auc	5	803	747	724	706	6/15		1:f-auc	675	∞	∞	∞	∞	0/15	
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15		f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15	
0:unif	2	2	4	6	7	15/15		0:unif	3	∞	∞	∞	∞	0/15	
1:f-auc	2	2	3	5	7	15/15		1:f-auc	2	440	∞	∞	∞	0/15	
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15		f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15	
0:unif	4	0.2	0.1	0.1	0.1	4/15		0:unif	∞	∞	∞	∞	∞	0/15	
1:f-auc	4	0.1	0.2	0.1	0.1	4/15		1:f-auc	∞	∞	∞	∞	∞	0/15	

Table 4: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:unif is uniform DE and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

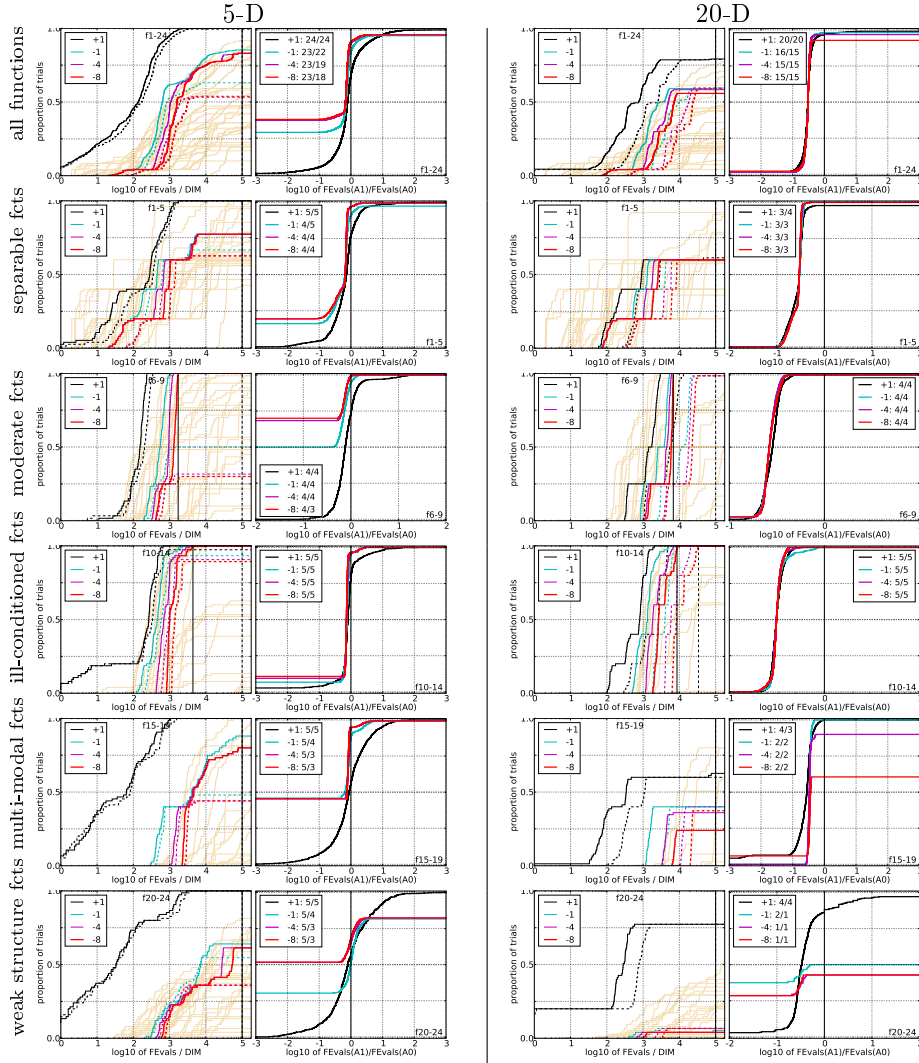
5.5 *F-AUC-Bandit* versus DE/rand/1

Figure 6: 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 F-AUC (solid) and DE/rand/1 (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 F-AUC divided by DE/rand/1, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:DE1	13	100	190	288	374	15/15	0:DE1	299	862	1419	1969	2543	15/15
1:f-auc	6	67 *2	132 *3	203 *3	266 *3	15/15	1:f-auc	93 *3	265 *3	431 *3	597 *3	763 *3	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:DE1	23	36	48	59	70	15/15	0:DE1	149	211	272	333	395	15/15
1:f-auc	18 *3	26 *3	35 *3	42 *3	50 *3	15/15	1:f-auc	48 *3	68 *3	86 *3	105 *3	123 *3	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:DE1	4	849	1226	1980	1976	2/15	0:DE1	5709	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	3	59	60	60 *	60 *	13/15	1:f-auc	∞	∞	∞	∞	∞ 2.0e6	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:DE1	6	4156	∞	∞	∞ 5.0e5	0/15	0:DE1	∞	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	6	∞	∞	∞	∞ 5.0e5	0/15	1:f-auc	∞	∞	∞	∞	∞ 2.0e6	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:DE1	35	72	73	73	73	15/15	0:DE1	230	266	267	267	267	15/15
1:f-auc	15 *2	24 *3	24 *3	24 *3	24 *3	15/15	1:f-auc	42 *3	53 *3	54 *3	54 *3	54 *3	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:DE1	9	1569	1.2e4	∞	∞ 5.0e5	0/15	0:DE1	73	56	56	58	59	15/15
1:f-auc	6	8 *3	7 *3	5 *3	5 *3	15/15	1:f-auc	19 *3	14 *3	14 *3	14 *3	14 *3	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:DE1	15	1	2	2	2	15/15	0:DE1	17	6	5	5	5	15/15
1:f-auc	13	1*	1*3	1*3	1*3	15/15	1:f-auc	5 *3	2 *3	2 *3	2 *3	2 *3	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:DE1	16	9673	1.8e4	1.7e4	1.7e4	1/15	0:DE1	80	121	125	126	129	14/15
1:f-auc	13	11 *3	11 *3	13 *3	14 *3	15/15	1:f-auc	23 *3	23 *3	24 *3	25 *3	26 *3	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:DE1	34	4694	1.1e4	9712	8830	2/15	0:DE1	99	111	115	117	119	15/15
1:f-auc	24	16 *2	16 *3	16 *3	17 *3	15/15	1:f-auc	26 *3	27 *3	29 *3	30 *3	31 *3	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:DE1	5	5	7	6	7	15/15	0:DE1	8	7	7	8	9	15/15
1:f-auc	4 *	4 *3	5 *3	5 *3	5 *3	15/15	1:f-auc	2 *3	2 *3	2 *3	2 *3	3 *3	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:DE1	7	3	3	3	3	15/15	0:DE1	21	7	7	8	8	15/15
1:f-auc	6	2 *3	2 *3	2 *3	2 *3	15/15	1:f-auc	7 *3	2 *3	2 *3	2 *3	2 *3	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:DE1	742	687	961	444	388	7/15	0:DE1	86	49	60	32	37	15/15
1:f-auc	22 *2	13	14	6	7	15/15	1:f-auc	27 *3	19 *3	20 *3	9 *3	10 *3	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:DE1	13	15	4	5	4	15/15	0:DE1	82	37	8	8	8	15/15
1:f-auc	10 *3	11 *3	3 *3	3 *3	3 *3	15/15	1:f-auc	25 *3	11 *3	2 *3	2 *3	3 *3	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:DE1	2	21	18	15	11	15/15	0:DE1	110	125	73	59	8	15/15
1:f-auc	2	15 *3	13 *3	11 *3	8 *3	15/15	1:f-auc	33 *3	38 *3	23 *3	19 *3	3 *3	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:DE1	7	362	350	∞	∞ 5.0e5	0/15	0:DE1	∞	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	5	7 *2	7 *2	7 *2	7 *2	12/15	1:f-auc	474	∞	∞	∞	∞ 2.0e6	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:DE1	9	382	193	173	167	3/15	0:DE1	∞	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞ 2.0e6	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:DE1	2	3	2	2	2	15/15	0:DE1	87	19	6	5	9	13/15
1:f-auc	6	2 *2	1*3	1*3	1*3	15/15	1:f-auc	23 *3	7 *3	2 *3	2 *3	5 *	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:DE1	5	0.9	0.8	1	1	15/15	0:DE1	35	5	3	3	3	15/15
1:f-auc	4	0.8 *2	0.7 *3	0.9 *3	1.0 *3	15/15	1:f-auc	11 *3	2 *3	3	11	28	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:DE1	49	∞	∞	∞	∞ 5.0e5	0/15	0:DE1	5830	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	29	1726 *3	11 *3	11 *3	10 *3	5/15	1:f-auc	1315 *3	∞	∞	∞	∞ 2.0e6	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:DE1	11	5	5	5	5	10/15	0:DE1	144	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40 *3	∞	∞	∞	∞ 2.0e6	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:DE1	4	110	109	107	106	11/15	0:DE1	35	286	276	261	231	5/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12 *3	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:DE1	6	612	747	724	706	6/15	0:DE1	701	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675 *	∞	∞	∞	∞ 2.0e6	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:DE1	2	11	225	∞	∞ 5.0e5	0/15	0:DE1	3	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	2	2	3 *3	5 *3	7 *3	15/15	1:f-auc	2	440 *2	∞	∞	∞ 2.0e6	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:DE1	6	∞	∞	∞	∞ 5.0e5	0/15	0:DE1	∞	∞	∞	∞	∞ 2.0e6	0/15
1:f-auc	4	0.1 *2	0.2 *2	0.1 *2	0.1 *2	4/15	1:f-auc	∞	∞	∞	∞	∞ 2.0e6	0/15

Table 5: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 – f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:DE1 is DE/rand/1 and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

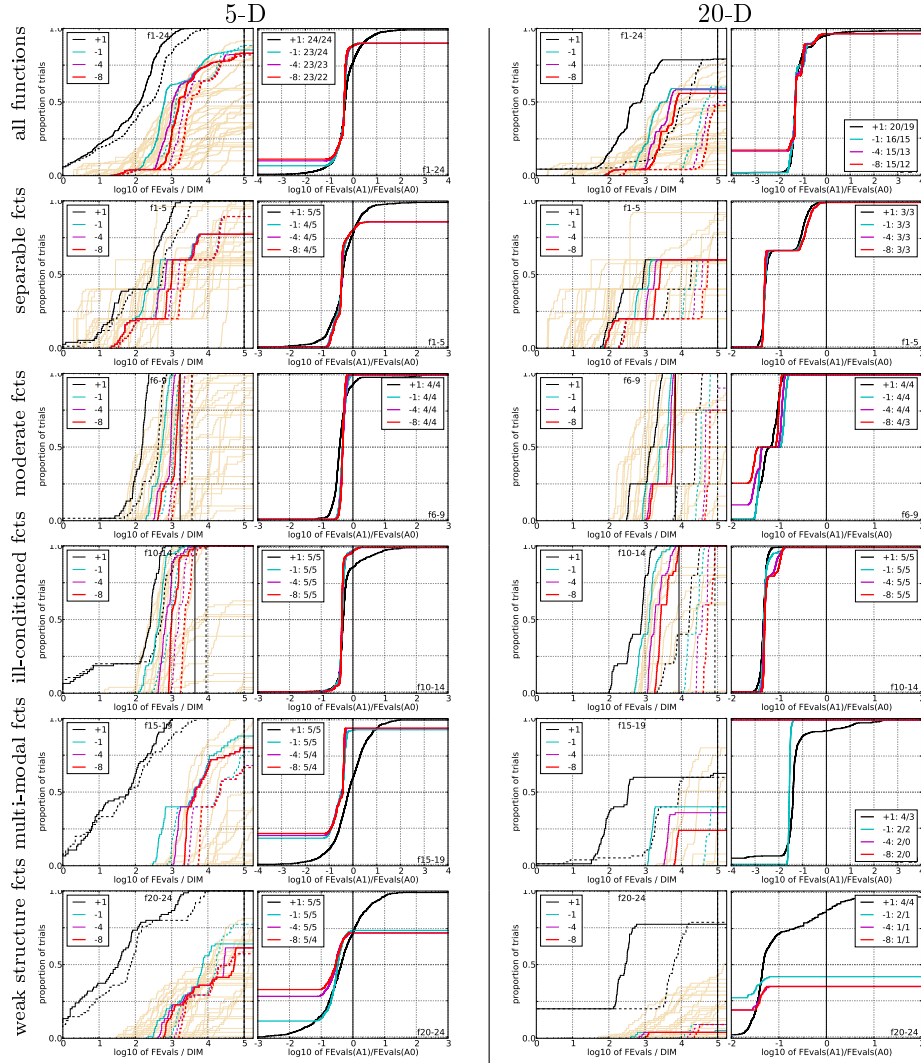
5.6 *F-AUC-Bandit* versus DE/rand/2

Figure 7: 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 F-AUC (solid) and DE/rand/2 (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 F-AUC divided by DE/rand/2, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:DE2	11	155	297	451	594	15/15	0:DE2	1820	5328	8748	1.2e4	1.6e4	15/15
1:f-auc	6	67*	132*	203*	266*	15/15	1:f-auc	93*	265*	431*	597*	763*	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:DE2	38	58	78	96	113	15/15	0:DE2	961	1337	1718	2109	2487	15/15
1:f-auc	18*	26*	35*	42*	50*	15/15	1:f-auc	48*	68*	86*	105*	123*	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:DE2	13	56	57	58	59	15/15	0:DE2	∞	∞	∞	∞	∞	0/15
1:f-auc	3*	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:DE2	11	401*	374*	362*	360*	7/15	0:DE2	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:DE2	21	36	36	36	36	15/15	0:DE2	121	130	130	130	130	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42*	53*	54*	54*	54*	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:DE2	20	19	15	12	12	15/15	0:DE2	544	357	333	∞	∞	0/15
1:f-auc	6*	8*	7*	5*	5*	15/15	1:f-auc	19*	14*	14*	14*	14*	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:DE2	23	3	3	3	3	15/15	0:DE2	105	55	52	52	54	15/15
1:f-auc	13	1*3	1*3	1*3	1*3	15/15	1:f-auc	5*3	2*3	2*3	2*3	2*3	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:DE2	33	20	22	25	29	15/15	0:DE2	243	170	198	225	253	15/15
1:f-auc	13*	11*	11*	13*	14*	15/15	1:f-auc	23*	23*	24*	25*	26*	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:DE2	63	31	27	30	32	15/15	0:DE2	280	207	240	272	303	15/15
1:f-auc	24*	16*	16*	16*	17*	15/15	1:f-auc	26*	27*	29*	30*	31*	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:DE2	9	9	11	11	12	15/15	0:DE2	50	49	45	48	56	15/15
1:f-auc	4*	4*	5*	5*	5*	15/15	1:f-auc	2*	2*	2*	2*	3*	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:DE2	11	4	4	5	5	15/15	0:DE2	144	47	46	49	51	15/15
1:f-auc	6*	2*	2*	2*	2*	15/15	1:f-auc	7*	2*	2*	2*	2*	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:DE2	58	29	30	13	13	15/15	0:DE2	577	288	239	95	97	15/15
1:f-auc	22*	13*	14*	6*	7*	15/15	1:f-auc	27*	19*	20*	9*	10*	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:DE2	20	24	7	7	7	15/15	0:DE2	507	227	49	49	50	15/15
1:f-auc	10*	11*	3*	3*	3*	15/15	1:f-auc	25*	11*	2*	2*	3*	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:DE2	1	34	30	26	18	15/15	0:DE2	787	841	487	397	55	15/15
1:f-auc	2	15*	13*	11*	8*	15/15	1:f-auc	33*	38*	23*	19*	3*	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:DE2	17	7	7	7	7	14/15	0:DE2	∞	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474*	∞	∞	∞	∞	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:DE2	12	226	92	83	94	6/15	0:DE2	∞	∞	∞	∞	∞	0/15
1:f-auc	6	31*	12*	11*	10*	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:DE2	6	5	3	3	3	15/15	0:DE2	352	227	∞	∞	∞	0/15
1:f-auc	6	2*	1*3	1*3	1*3	15/15	1:f-auc	23	7*	2*	2*	5*	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:DE2	7	2	2	2	2	15/15	0:DE2	281	64	∞	∞	∞	0/15
1:f-auc	4	0.8*	0.7*	0.9*	1.0*	15/15	1:f-auc	11*	2*	3*	11*	28*	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:DE2	74	6956	∞	∞	∞	0/15	0:DE2	3.7e4	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315*	∞	∞	∞	∞	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:DE2	22	2	1	1	1	15/15	0:DE2	937	∞	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40*	∞	∞	∞	∞	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:DE2	7	26	27	27	27	14/15	0:DE2	370	183	181	176	159	7/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12*	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:DE2	8	474	442	429	419	8/15	0:DE2	614	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:DE2	3	7	12	∞	∞	0/15	0:DE2	2	∞	∞	∞	∞	0/15
1:f-auc	2	2*	3*	5*	7*	15/15	1:f-auc	2	440*	∞	∞	∞	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:DE2	16	0.2	0.1	0.1	0.1	6/15	0:DE2	∞	∞	∞	∞	∞	0/15
1:f-auc	4*	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 6: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:DE2 is DE/rand/2 and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

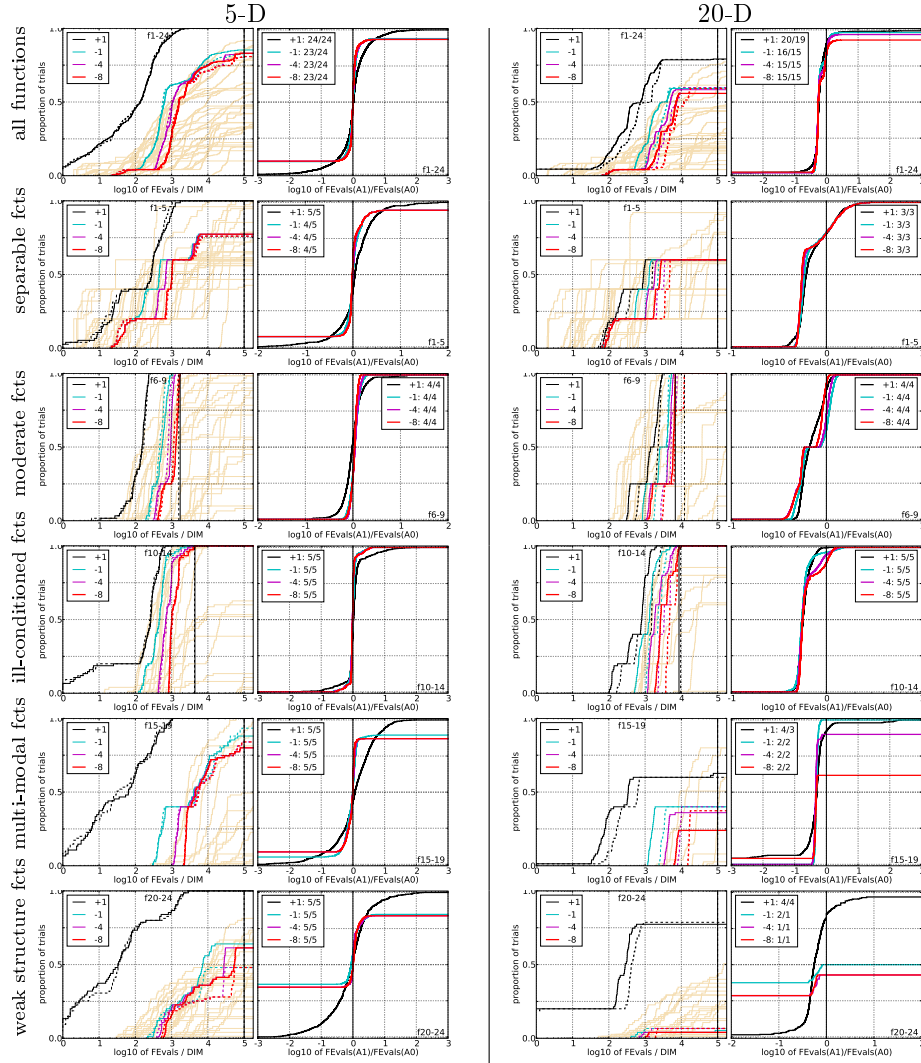
5.7 *F-AUC-Bandit* versus DE/rand-to-best/2

Figure 8: 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 F-AUC (solid) and DE/rand-to-best/2 (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 F-AUC divided by DE/rand-to-best/2, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:DE3	6	69	133	202	272	15/15	0:DE3	166	489	818	1139	1465	15/15
1:f-auc	6	67	132	203	266	15/15	1:f-auc	93*3	265*3	431*3	597*3	763*3	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:DE3	17	26	34	43	51	15/15	0:DE3	83	118	154	190	225	15/15
1:f-auc	18	26	35	42	50	15/15	1:f-auc	48*3	68*3	86*3	105*3	123*3	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:DE3	4	166	166	166	166	10/15	0:DE3	∞	∞	∞	∞	∞	0/15
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:DE3	4	1943	1805	1740	1725	2/15	0:DE3	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:DE3	11	18	18	18	18	15/15	0:DE3	37	47	47	47	47	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42	53	54	54	54	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:DE3	6	8	6*	5	5*	15/15	0:DE3	33	25	24	25	25	15/15
1:f-auc	6	8	7	5	5	15/15	1:f-auc	19*3	14*3	14*3	14*3	14*3	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:DE3	12	1	1	1	1	15/15	0:DE3	9	4	3	3	4	15/15
1:f-auc	13	1	1	1	1	15/15	1:f-auc	5*3	2*3	2*3	2*3	2*3	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:DE3	13	9	10	11	13	15/15	0:DE3	26	21	23	25	28	15/15
1:f-auc	13	11	11	13	14	15/15	1:f-auc	23	23	24	25	26	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:DE3	28	14	13	14	15	15/15	0:DE3	30	26	28	31	33	15/15
1:f-auc	24	16	16	16	17	15/15	1:f-auc	26*	27	29	30	31	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:DE3	4	4	5	5	5	15/15	0:DE3	4	4	4	4	5	15/15
1:f-auc	4	4	5	5	5	15/15	1:f-auc	2*3	2*3	2*3	2*3	3*3	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:DE3	6	2	2	2	3	15/15	0:DE3	11	4	4	4	5	15/15
1:f-auc	6	2	2	2	2	15/15	1:f-auc	7*3	2*3	2*3	2*3	2*3	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:DE3	22	13	14	6	6	15/15	0:DE3	49	27	24	10	10	15/15
1:f-auc	22	13	14	6	7	15/15	1:f-auc	27*3	19	20	9	10	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:DE3	10	11	3	3	3	15/15	0:DE3	47	21	5	5	5	15/15
1:f-auc	10	11	3	3	3	15/15	1:f-auc	25*3	11*3	2*3	2*3	3*3	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:DE3	2	15	14	12	8	15/15	0:DE3	54	72	42	34	5	15/15
1:f-auc	2	15	13	11	8	15/15	1:f-auc	33*3	38*3	23*3	19*3	3*3	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:DE3	5	7	7	7	7	12/15	0:DE3	∞	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474*2	∞	∞	∞	∞	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:DE3	7	26	10	9	9	14/15	0:DE3	∞	∞	∞	∞	∞	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:DE3	3	2	1	1	1	15/15	0:DE3	41	13	4	4	6	14/15
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23	7*3	2*3	2*3	5*	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:DE3	3	0.7	0.6	0.8	1.0	15/15	0:DE3	20	4	2	2	3	14/15
1:f-auc	4	0.8	0.7	0.9	1.0	15/15	1:f-auc	11*3	2*3	3	11	28	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:DE3	44	1209	7	7	7	7/15	0:DE3	3223	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315*3	∞	∞	∞	∞	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:DE3	9	26	19	18	18	5/15	0:DE3	85	∞	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40*3	∞	∞	∞	∞	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:DE3	5	263	258	255	251	8/15	0:DE3	19	285	275	259	230	5/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12*3	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:DE3	4	802	746	724	705	6/15	0:DE3	331	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:DE3	2	2	3	5	6	15/15	0:DE3	2	∞	∞	∞	∞	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440*2	∞	∞	∞	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:DE3	4	0.5	0.3	0.3	0.3	2/15	0:DE3	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 7: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:DE3 is DE/rand-to-best/2 and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

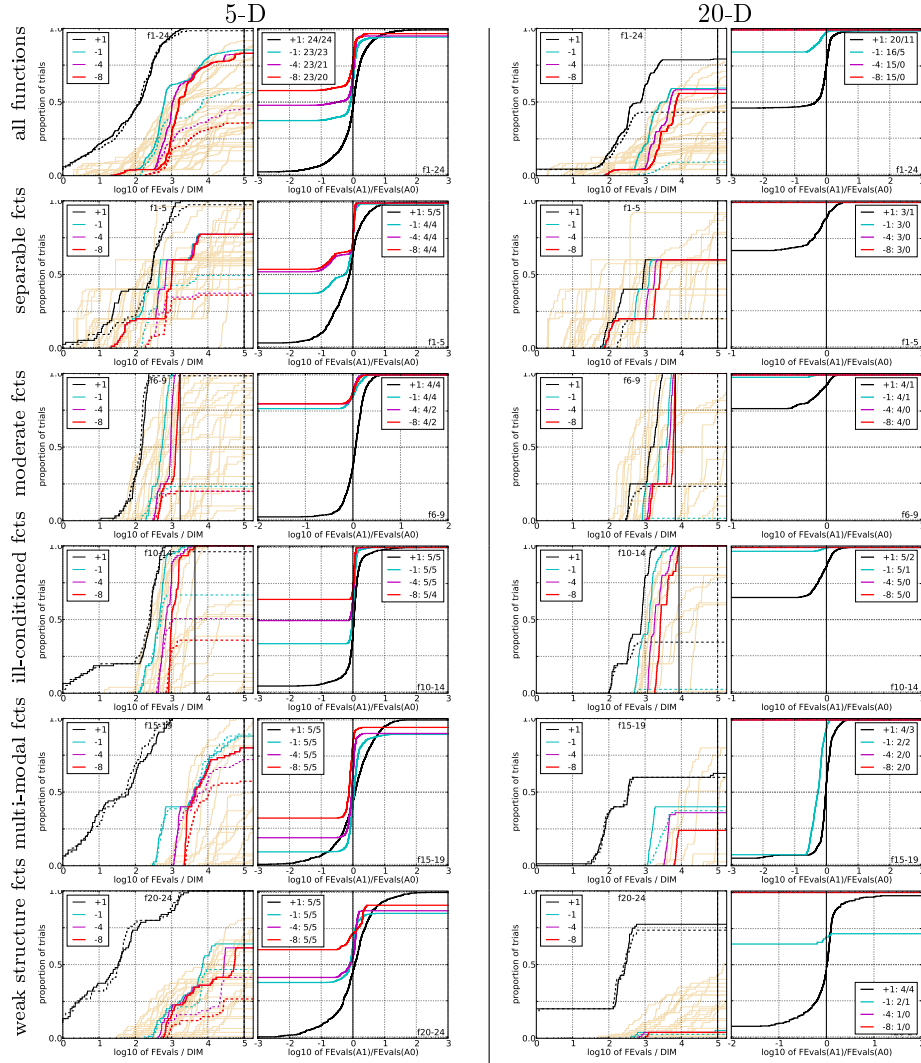
5.8 *F-AUC-Bandit* versus DE/current-to-rand/1

Figure 9: 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 *F-AUC* (solid) and *DE/current-to-rand/1* (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 *F-AUC* divided by *DE/current-to-rand/1*, 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 (*F-AUC* first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:DE4	9	3001	1.0e4	1.5e4	1.5e4	11/15	0:DE4	114	∞	∞	∞	∞	0/15
1:f-auc	6	67	132	203	266	15/15	1:f-auc	93	265*3	431*3	597*3	763*3	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:DE4	19	2083	3746	4799	6113	7/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	18	26	35	42	50	15/15	1:f-auc	48*3	68*3	86*3	105*3	123*3	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:DE4	3	623	1228	1983	1980	2/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	3	59	60	60*	60*2	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:DE4	103	∞	∞	∞	∞	0/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:DE4	91	5.7e4	5.7e4	5.7e4	5.7e4	7/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	15*3	24*3	24*3	24*3	24*3	15/15	1:f-auc	42*3	53*3	54*3	54*3	54*3	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:DE4	319	2.5e4	∞	∞	∞	0/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	6	8*3	7*3	5*3	5*3	15/15	1:f-auc	19*3	14*3	14*3	14*3	14*3	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:DE4	13	156	117	117	115	11/15	0:DE4	112	2949	∞	∞	∞	0/15
1:f-auc	13	1	1	1	1	15/15	1:f-auc	5	2*3	2*3	2*3	2*3	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:DE4	11	2.1e4	∞	∞	∞	0/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	13	11*3	11*3	13*3	14*3	15/15	1:f-auc	23*3	23*3	24*3	25*3	26*3	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:DE4	20	3.3e4	2.3e4	2.1e4	1.9e4	1/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	24	16*2	16*2	16*2	17*2	15/15	1:f-auc	26*3	27*3	29*3	30*3	31*3	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:DE4	4	584	704	532	502	8/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	4	4	5	5	5	15/15	1:f-auc	2*3	2*3	2*3	2*3	3*3	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:DE4	5	103	108	126	201	9/15	0:DE4	735	∞	∞	∞	∞	0/15
1:f-auc	6	2	2	2	2	15/15	1:f-auc	7	2*3	2*3	2*3	2*3	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:DE4	353	3719	1.5e4	5379	∞	0/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	22	13	14*3	6*3	7*3	15/15	1:f-auc	27*3	19*3	20*3	9*3	10*3	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:DE4	591	1011	766	1145	890	3/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	10	11	3	3	3	15/15	1:f-auc	25*3	11*3	2*3	2*3	3*3	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:DE4	2	629	271	508	927	7/15	0:DE4	34	4.3e4	∞	∞	∞	0/15
1:f-auc	2	15	13	11	8	15/15	1:f-auc	33	38*3	23*3	19*3	3*3	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:DE4	4	14	18	17	28	7/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:DE4	3	15	12	16	26	9/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:DE4	6	2	1	1	2	13/15	0:DE4	22	13	426	∞	∞	0/15
1:f-auc	6	2	1	1	1*	15/15	1:f-auc	23	7*2	2*3	2*3	5*3	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:DE4	4	10	9	8	7	13/15	0:DE4	11	19	∞	∞	∞	0/15
1:f-auc	4	0.8	0.7	0.9	1.0*	15/15	1:f-auc	11	2*3	3*3	11*3	28*3	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:DE4	43	771	11	20	30	1/15	0:DE4	1337	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315	∞	∞	∞	∞	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:DE4	10	86	60	60	59	2/15	0:DE4	7	36*	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40	∞	∞	∞	∞	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:DE4	3	151	259	256	328	7/15	0:DE4	12	923	1913	∞	∞	0/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:DE4	3	802	1368	1927	3050	2/15	0:DE4	2157	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:DE4	2	2	3*3	7	13	8/15	0:DE4	3	∞	∞	∞	∞	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440	∞	∞	∞	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:DE4	3	0.5	0.3	0.3	0.6	1/15	0:DE4	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 8: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $ERT(10^{-7}) = \infty$. 0:DE4 is DE/current-to-rand/1 and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

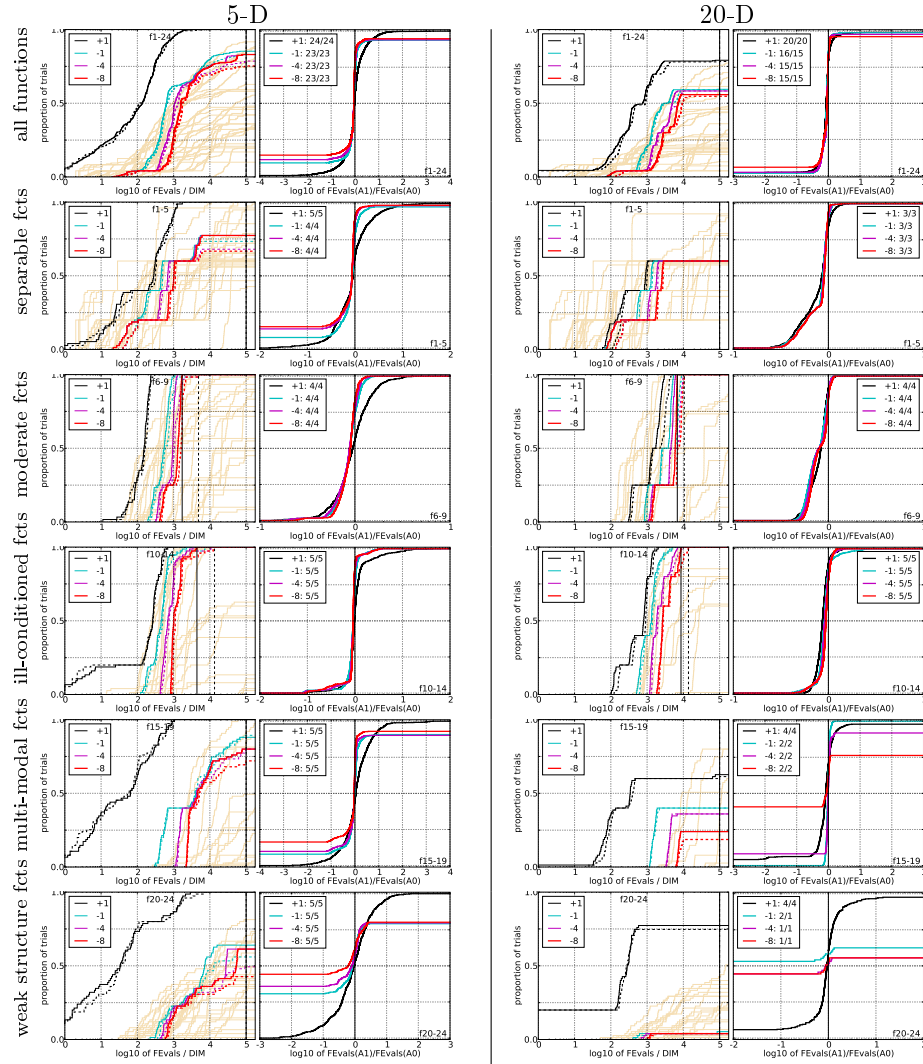
5.9 *F-AUC-Bandit* versus PM-AdapSS-DE

Figure 10: 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 F-AUC (solid) and PM-AdapSS-DE (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 F-AUC divided by PM-AdapSS-DE, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:pmDE	5	74	145	223	292	15/15	0:pmDE	101	291	465	646	827	15/15
1:f-auc	6	67	132	203*	266*2	15/15	1:f-auc	93	265*	431	597*	763*2	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:pmDE	19	28	38	47	55	15/15	0:pmDE	52	73	93	112	132	15/15
1:f-auc	18	26*2	35*2	42*2	50*3	15/15	1:f-auc	48*	68*	86*2	105*2	123*	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:pmDE	5	166	362	621	621	5/15	0:pmDE	∞	∞	∞	∞	∞	0/15
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:pmDE	5	∞	∞	∞	∞	0/15	0:pmDE	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:pmDE	24	36	36	36	36	15/15	0:pmDE	82	96	96	96	96	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42*3	53*3	54*3	54*3	54*3	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:pmDE	8	9	7	6	6	15/15	0:pmDE	20	15	14	14	14	15/15
1:f-auc	6	8	7	5	5	15/15	1:f-auc	19	14	14	14	14	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:pmDE	12	1	1	1	1	15/15	0:pmDE	6	2	2	2	2	15/15
1:f-auc	13	1	1	1	1	15/15	1:f-auc	5*2	2	2	2	2	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:pmDE	13	15	19	20	21	15/15	0:pmDE	35	38	39	40	41	15/15
1:f-auc	13	11	11*2	13*2	14*2	15/15	1:f-auc	23*3	23*3	24*3	25*3	26*3	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:pmDE	25	21	21	21	22	15/15	0:pmDE	37	43	45	46	47	15/15
1:f-auc	24	16	16*	16*	17*	15/15	1:f-auc	26*3	27*3	29*3	30*3	31*3	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:pmDE	5	4	5	5	6	15/15	0:pmDE	3	3	2	3	3	15/15
1:f-auc	4	4	5	5*	5*2	15/15	1:f-auc	2*	2*2	2*2	2*3	3*	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:pmDE	6	2	2	2	3	15/15	0:pmDE	9	3	3	3	3	15/15
1:f-auc	6	2	2*2	2*	2*3	15/15	1:f-auc	7*	2	2*	2*	2*	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:pmDE	24	24	27	12	13	15/15	0:pmDE	29	20	24	12	13	15/15
1:f-auc	22	13	14	6	7	15/15	1:f-auc	27	19	20	9	10*	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:pmDE	10	12	3	3	3	15/15	0:pmDE	28	12	3	3	3	15/15
1:f-auc	10	10	3*2	3*2	3*2	15/15	1:f-auc	25*2	11*2	2*	2	3*	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:pmDE	1	17	15	12	9	15/15	0:pmDE	43	43	25	20	3	15/15
1:f-auc	2	15	13*	11*	8*	15/15	1:f-auc	33*2	38*2	23	19	3	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:pmDE	5	3	3	8	7	12/15	0:pmDE	∞	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:pmDE	4	55	20	28	27	11/15	0:pmDE	2.2e4	∞	∞	∞	∞	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:pmDE	4	2	1	1	1	15/15	0:pmDE	23	7	2	2	18	7/15
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23	7	2	2	5	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:pmDE	4	0.8	0.7	0.9	1	15/15	0:pmDE	13	2	8	6	8	7/15
1:f-auc	4	0.8	0.7	0.9	1.0	15/15	1:f-auc	11	2	3	11	28	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:pmDE	37	2054	19	19	29	1/15	0:pmDE	1839	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315	∞	∞	∞	∞	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:pmDE	11	9	7	7	7	8/15	0:pmDE	46	∞	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40	∞	∞	∞	∞	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:pmDE	4	201	198	195	192	9/15	0:pmDE	12	568	547	515	456	3/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:pmDE	4	469	437	424	414	8/15	0:pmDE	1572	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:pmDE	2	12	14	21	27	5/15	0:pmDE	2	∞	∞	∞	∞	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440	∞	∞	∞	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:pmDE	6	0.2	0.4	0.3	0.3	2/15	0:pmDE	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 9: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOb-2009 (given in the respective first row) for different Δf values for functions f_1 – f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $ERT(10^{-7}) = \infty$. 0:pmDE is PM-AdapSS-DE and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

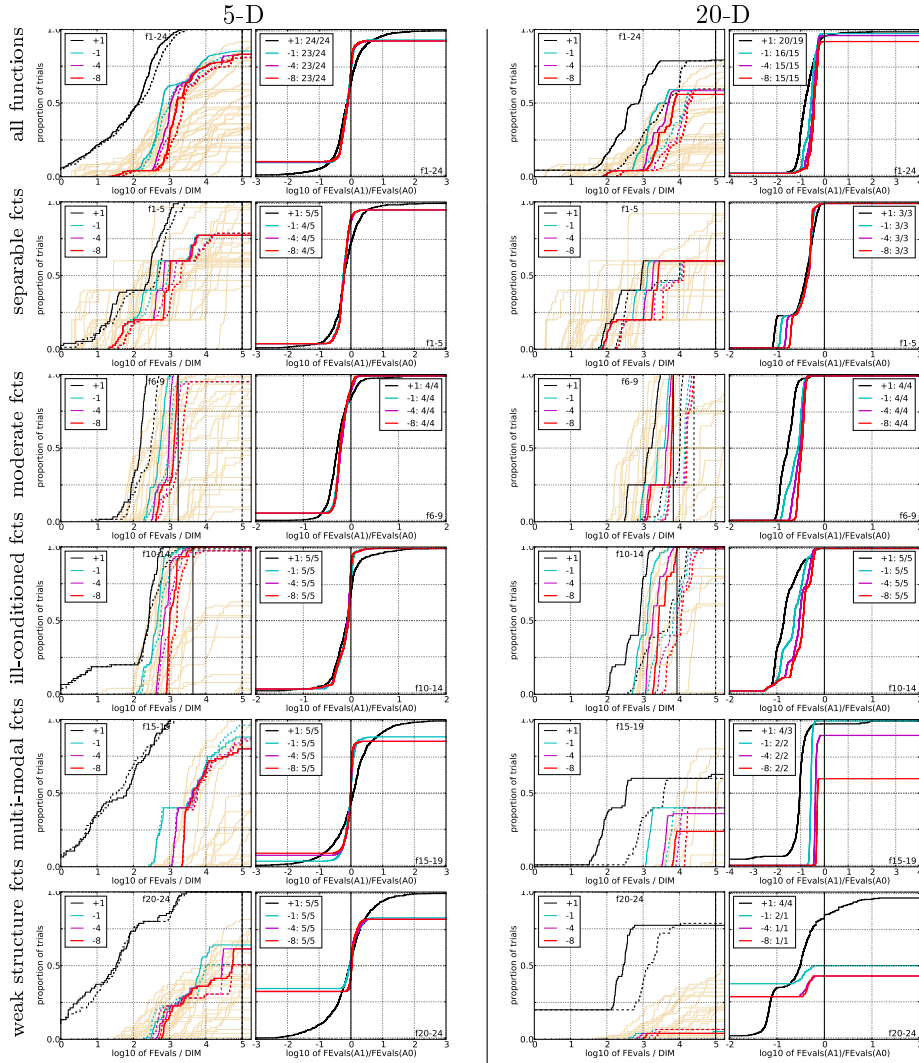
5.10 *F-AUC-Bandit* versus DMAB

Figure 11: 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 F-AUC (solid) and DMAB (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 F-AUC divided by DMAB, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:dmab	10	102	193	284	370	15/15	0:dmab	155	494	816	1145	1471	15/15
1:f-auc	6	67*	132*²	203*²	266*²	15/15	1:f-auc	93*³	265*³	431*³	597*³	763*³	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:dmab	32	48	65	79	93	15/15	0:dmab	390	430	465	501	536	15/15
1:f-auc	18*³	26*³	35*³	42*³	50*³	15/15	1:f-auc	48*³	68*³	86*³	105*³	123*³	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:dmab	7	75	75	76	76	13/15	0:dmab	∞	∞	∞	∞	∞	0/15
1:f-auc	3	59*	60*	60*	60*	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:dmab	8	4173	3876	3736	3702	1/15	0:dmab	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:dmab	21	37	37	37	37	15/15	0:dmab	113	124	124	124	124	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42*³	53*³	54*³	54*³	54*³	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:dmab	14	289	144	83	67	13/15	0:dmab	154	98	71	60	52	15/15
1:f-auc	6*	8*²	7*	5*	5*	15/15	1:f-auc	19*³	14*³	14*³	14*³	14*³	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:dmab	19	2	2	2	2	15/15	0:dmab	41	8	6	6	6	15/15
1:f-auc	13	1	1	1	1*	15/15	1:f-auc	5*³	2*³	2*³	2*³	2*³	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:dmab	25	124	110	109	109	14/15	0:dmab	108	70	71	71	72	15/15
1:f-auc	13*²	11*²	11*³	13*³	14*³	15/15	1:f-auc	23*³	23*³	24*³	25*³	26*³	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:dmab	47	25	23	25	26	15/15	0:dmab	122	83	83	83	84	15/15
1:f-auc	24*	16*²	16*	16*²	17*²	15/15	1:f-auc	26*³	27*³	29*³	30*³	31*³	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:dmab	7	6	8	8	9	15/15	0:dmab	24	20	17	17	19	15/15
1:f-auc	4*²	4*³	5*³	5*³	5*³	15/15	1:f-auc	2*³	2*³	2*³	2*³	3*³	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:dmab	7	3	3	3	3	15/15	0:dmab	32	10	10	11	11	15/15
1:f-auc	6	2	2*²	2	2*	15/15	1:f-auc	7*³	2*³	2*³	2*³	2*³	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:dmab	372	227	189	69	62	13/15	0:dmab	415	169	121	44	41	14/15
1:f-auc	22*²	13	14	6	7	15/15	1:f-auc	27*³	19*³	20*³	9*³	10*³	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:dmab	10	11	3	3	3	15/15	0:dmab	172	50	9	8	7	15/15
1:f-auc	10	11	3	3	3	15/15	1:f-auc	25*³	11*³	2*³	2*³	3*³	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:dmab	2	17	14	12	9	15/15	0:dmab	187	107	51	39	5	15/15
1:f-auc	2	15	13	11	8	15/15	1:f-auc	33*³	38*³	23*³	19*³	3*³	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:dmab	6	3	3	3	3	14/15	0:dmab	∞	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474*	∞	∞	∞	∞	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:dmab	5	24	14	12	12	13/15	0:dmab	∞	∞	∞	∞	∞	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:dmab	3	2	1	1	1	15/15	0:dmab	254	24	5	4	3	15/15
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23*²	7*³	2*³	2*³	5	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:dmab	3	0.7	0.6	0.8	4	14/15	0:dmab	124	7	3	2	2	15/15
1:f-auc	4	0.8	0.7	0.9	1.0	15/15	1:f-auc	11*³	2*³	3	11	28	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:dmab	50	1129	6	6	6	8/15	0:dmab	1.4e4	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315*³	∞	∞	∞	∞	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:dmab	8	36	25	25	25	4/15	0:dmab	583	∞	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40*³	∞	∞	∞	∞	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:dmab	5	200	197	194	191	9/15	0:dmab	59	286	276	260	230	5/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12*³	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:dmab	5	469	436	423	413	8/15	0:dmab	408	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:dmab	2	3	3	5	7	15/15	0:dmab	2	∞	∞	∞	∞	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440*²	∞	∞	∞	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:dmab	5	0.5	0.3	0.3	0.3	2/15	0:dmab	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 10: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 – f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:dmab is DMAB and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

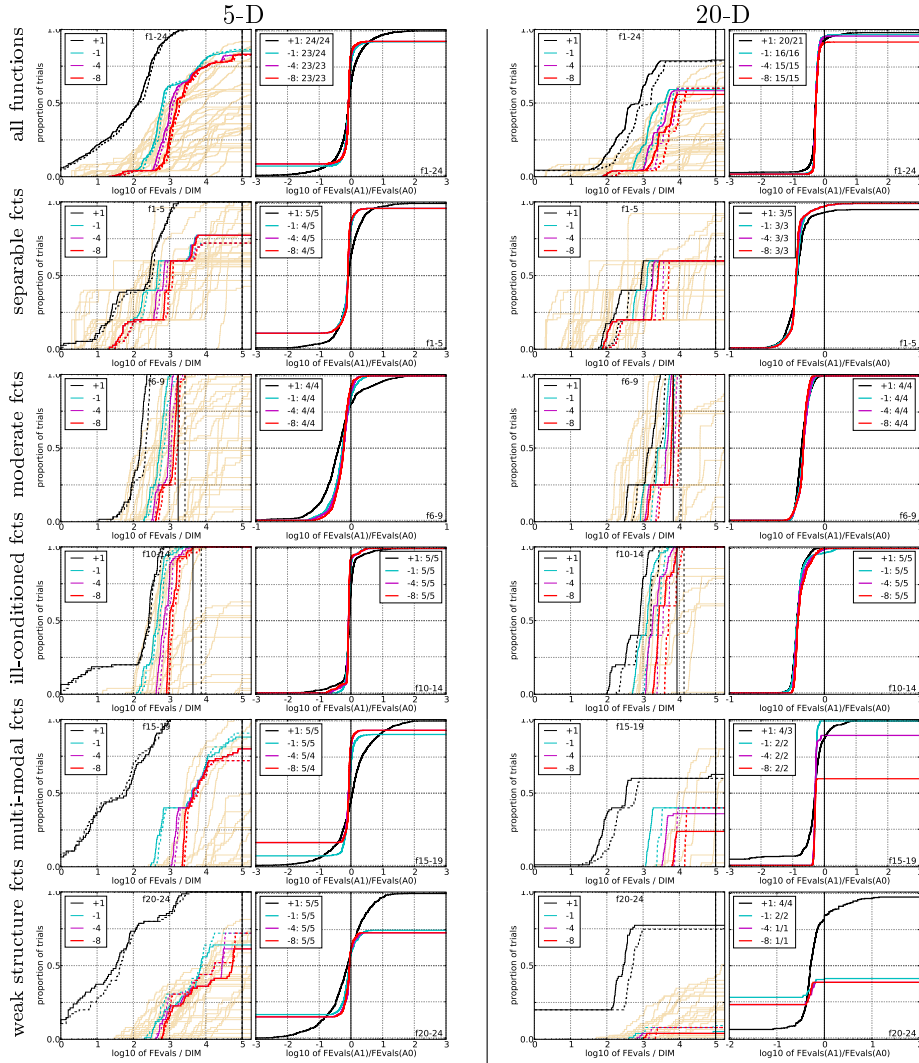
5.11 *F-AUC-Bandit* versus AP

Figure 12: 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 F-AUC (solid) and AP (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 F-AUC divided by AP, 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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f_1	11	12	12	12	12	15/15	f_1	43	43	43	43	43	15/15
0:ap	6	85	171	242	324	15/15	0:ap	178	520	858	1203	1543	15/15
1:f-auc	6	67	132*3	203*3	266*3	15/15	1:f-auc	93*3	265*3	431*3	597*3	763*3	15/15
f_2	83	88	90	92	94	15/15	f_2	385	387	390	391	393	15/15
0:ap	22	31	41	50	59	15/15	0:ap	90	128	166	203	240	15/15
1:f-auc	18*2	26*3	35*3	42*3	50*3	15/15	1:f-auc	48*3	68*3	86*3	105*3	123*3	15/15
f_3	716	1637	1646	1650	1654	15/15	f_3	5066	7635	7643	7646	7651	15/15
0:ap	4	282	281	281	281	8/15	0:ap	5901	∞	∞	∞	∞	0/15
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_4	809	1688	1817	1886	1903	15/15	f_4	4722	7666	7700	7758	1.4e5	9/15
0:ap	5	4168	3872	3732	3699	1/15	0:ap	6315	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_5	10	10	10	10	10	15/15	f_5	41	41	41	41	41	15/15
0:ap	22	37	37	37	37	15/15	0:ap	87	103	103	103	103	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42*2	53*2	54*2	54*2	54*2	15/15
f_6	114	281	580	1038	1332	15/15	f_6	1296	3413	5220	6728	8409	15/15
0:ap	9	10	8	6	6	15/15	0:ap	33	24	23	23	23	15/15
1:f-auc	6	8*2	7*2	5*3	5*3	15/15	1:f-auc	19*3	14*3	14*3	14*3	14*3	15/15
f_7	24	1171	1572	1572	1597	15/15	f_7	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:ap	15	1	1	1	2	15/15	0:ap	4	3	3	3	3	15/15
1:f-auc	13	1	1	1	1*	15/15	1:f-auc	5*3	2*3	2*3	2*3	2*3	15/15
f_8	73	336	391	410	422	15/15	f_8	2039	4040	4219	4371	4484	15/15
0:ap	16	14	17	18	20	15/15	0:ap	36	34	36	38	41	15/15
1:f-auc	13*2	11	11*2	13*2	14*3	15/15	1:f-auc	23*3	23*3	24*3	25*3	26*3	15/15
f_9	35	214	300	335	369	15/15	f_9	1716	3277	3455	3594	3727	15/15
0:ap	35	22	20	21	22	15/15	0:ap	44	43	45	47	49	15/15
1:f-auc	24*2	16*	16	16*	17*2	15/15	1:f-auc	26*3	27*3	29*3	30*3	31*3	15/15
f_{10}	349	574	626	829	880	15/15	f_{10}	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:ap	5	5	6	5	6	15/15	0:ap	5	5	4	5	5	15/15
1:f-auc	4	4*	5*3	5*3	5*3	15/15	1:f-auc	2*3	2*3	2*3	2*3	3*3	15/15
f_{11}	143	763	1177	1467	1673	15/15	f_{11}	1002	6278	9762	1.2e4	1.5e4	15/15
0:ap	7	2	2	3	3	15/15	0:ap	13	4	4	5	5	15/15
1:f-auc	6	2*2	2*3	2*3	2*3	15/15	1:f-auc	7*3	2*3	2*3	2*3	2*3	15/15
f_{12}	108	371	461	1303	1494	15/15	f_{12}	1042	2740	4140	1.2e4	1.4e4	15/15
0:ap	27	13	16	8	9	15/15	0:ap	51	29	29	14	15	15/15
1:f-auc	22*	13	14	6	7	15/15	1:f-auc	27*3	19	20*2	9*3	10*3	15/15
f_{13}	132	250	1310	1752	2255	15/15	f_{13}	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:ap	11	13	4	4	4	15/15	0:ap	51	23	5	5	5	15/15
1:f-auc	10	11*3	3*3	3*3	3*3	15/15	1:f-auc	25*3	11*3	2*3	2*3	3*3	15/15
f_{14}	10	58	139	251	476	15/15	f_{14}	75	304	932	1648	1.6e4	15/15
0:ap	3	18	16	13	10	15/15	0:ap	65	74	44	36	5	15/15
1:f-auc	2	15	13*2	11*3	8*3	15/15	1:f-auc	33*3	38*3	23*3	19*3	3*3	15/15
f_{15}	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f_{15}	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:ap	5	3	3	3	3	14/15	0:ap	∞	∞	∞	∞	∞	0/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	0/15
f_{16}	120	2662	1.0e4	1.2e4	1.2e4	15/15	f_{16}	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:ap	3	19	29	26	25	10/15	0:ap	∞	∞	∞	∞	∞	0/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f_{17}	5	899	3669	6351	7934	15/15	f_{17}	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:ap	3	3	2	1	2	15/15	0:ap	43	12	4	4	3	15/15
1:f-auc	6	2*2	1*2	1*	1*3	15/15	1:f-auc	23	7*3	2*3	2*3	5	13/15
f_{18}	103	3968	9280	1.1e4	1.2e4	15/15	f_{18}	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:ap	5	0.8	0.8	1	1	15/15	0:ap	21	3	2	2	2	15/15
1:f-auc	4	0.8	0.7	0.9*3	1.0*3	15/15	1:f-auc	11*3	2*3	3	11	28	5/15
f_{19}	1	242	1.2e5	1.2e5	1.2e5	15/15	f_{19}	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:ap	35	2153	∞	∞	∞	0/15	0:ap	2743	∞	∞	∞	∞	0/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315*	∞	∞	∞	∞	0/15
f_{20}	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f_{20}	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:ap	12	7	5	5	5	10/15	0:ap	85	∞	∞	∞	∞	0/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40*3	∞	∞	∞	∞	0/15
f_{21}	41	1674	1705	1729	1757	14/15	f_{21}	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:ap	6	48	47	47	46	13/15	0:ap	22	214	207	195	173	6/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12*3	568	547	515	456	3/15
f_{22}	71	938	1008	1040	1068	14/15	f_{22}	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:ap	5	269	251	244	238	10/15	0:ap	1586	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f_{23}	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f_{23}	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:ap	2	3	4	6	8	15/15	0:ap	2	440	∞	∞	∞	0/15
1:f-auc	2	2	3*3	5*2	7*3	15/15	1:f-auc	2	440	∞	∞	∞	0/15
f_{24}	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f_{24}	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:ap	5	0.1	0.1	0.1	0.1	6/15	0:ap	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 11: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 – f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:ap is AP and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

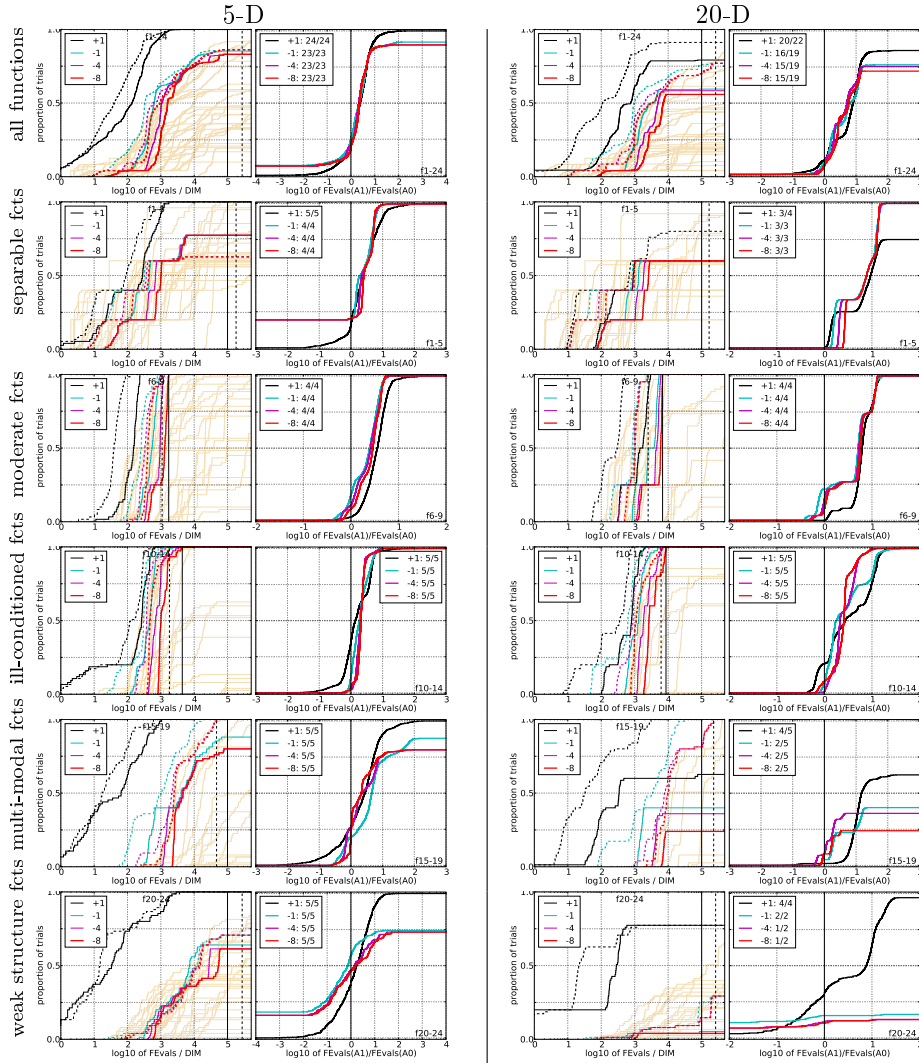
5.12 *F-AUC-Bandit* versus IPOP-CMA-ES

Figure 13: 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 F-AUC (solid) and IPOP-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 F-AUC divided by IPOP-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 (F-AUC first).

5-D							20-D						
Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ	Δf	1e+1	1e-1	1e-3	1e-5	1e-7	#succ
f₁	11	12	12	12	12	15/15	f₁	43	43	43	43	43	15/15
0:cma	2	14* ³	27* ³	39* ³	51* ³	15/15	0:cma	8* ³	20* ³	33* ³	46* ³	58* ³	15/15
1:f-auc	6	67	132	203	266	15/15	1:f-auc	93	265	431	597	763	15/15
f₂	83	88	90	92	94	15/15	f₂	385	387	390	391	393	15/15
0:cma	14* ³	18* ³	19* ³	21* ³	22* ³	15/15	0:cma	35* ³	43* ³	45* ³	47* ³	48* ³	15/15
1:f-auc	18	26	35	42	50	15/15	1:f-auc	48	68	86	105	123	15/15
f₃	716	1637	1646	1650	1654	15/15	f₃	5066	7635	7643	7646	7651	15/15
0:cma	2	3130	3113	3106	3099	2/15	0:cma	13* ³	∞	∞	∞	∞	0/15
1:f-auc	3	59	60	60	60	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f₄	809	1688	1817	1886	1903	15/15	f₄	4722	7666	7700	7758	1.4e5	9/15
0:cma	2*	∞	∞	∞	∞	0/15	0:cma	∞	∞	∞	∞	∞	0/15
1:f-auc	6	∞	∞	∞	∞	0/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f₅	10	10	10	10	10	15/15	f₅	41	41	41	41	41	15/15
0:cma	5* ³	6* ³	6* ³	6* ³	6* ³	15/15	0:cma	6* ³	7* ³	7* ³	7* ³	7* ³	15/15
1:f-auc	15	24	24	24	24	15/15	1:f-auc	42	53	54	54	54	15/15
f₆	114	281	580	1038	1332	15/15	f₆	1296	3413	5220	6728	8409	15/15
0:cma	2* ³	2* ³	2* ³	1* ³	1* ³	15/15	0:cma	2* ³	1* ³	1* ³	1* ³	1* ³	15/15
1:f-auc	6	8	7	5	5	15/15	1:f-auc	19	14	14	14	14	15/15
f₇	24	1171	1572	1572	1597	15/15	f₇	1351	9503	1.7e4	1.7e4	1.7e4	15/15
0:cma	4* ²	1	1	1	1	15/15	0:cma	2* ³	3	2	2	2	15/15
1:f-auc	13	1	1	1	1	15/15	1:f-auc	5	2*	2	2	2	15/15
f₈	73	336	391	410	422	15/15	f₈	2039	4040	4219	4371	4484	15/15
0:cma	3* ³	5* ²	6* ²	6* ³	6* ³	15/15	0:cma	4* ³	4* ³	4* ³	4* ³	4* ³	15/15
1:f-auc	13	11	11	13	14	15/15	1:f-auc	23	23	24	25	26	15/15
f₉	35	214	300	335	369	15/15	f₉	1716	3277	3455	3594	3727	15/15
0:cma	6* ³	9*	7* ³	7* ³	7* ³	15/15	0:cma	5* ³	6* ³	6* ³	6* ³	6* ³	15/15
1:f-auc	24	16	16	16	17	15/15	1:f-auc	26	27	29	30	31	15/15
f₁₀	349	574	626	829	880	15/15	f₁₀	7413	1.1e4	1.5e4	1.7e4	1.7e4	15/15
0:cma	4	3* ³	3* ³	2* ³	2* ³	15/15	0:cma	2* ³	2* ³	1* ³	1* ³	1* ³	15/15
1:f-auc	4	4	5	5	5	15/15	1:f-auc	2	2	2	2	3	15/15
f₁₁	143	763	1177	1467	1673	15/15	f₁₁	1002	6278	9762	1.2e4	1.5e4	15/15
0:cma	9	2	2* ³	1* ³	1* ³	15/15	0:cma	11	2* ²	1* ³	1* ³	1* ³	15/15
1:f-auc	6* ²	2	2	2	2	15/15	1:f-auc	7* ³	2	2	2	2	15/15
f₁₂	108	371	461	1303	1494	15/15	f₁₂	1042	2740	4140	1.2e4	1.4e4	15/15
0:cma	9* ²	6*	6*	3*	3*	15/15	0:cma	5* ³	6* ³	5* ³	2* ³	2* ³	15/15
1:f-auc	22	13	14	6	7	15/15	1:f-auc	27	19	20	9	10	15/15
f₁₃	132	250	1310	1752	2255	15/15	f₁₃	652	2751	1.9e4	2.4e4	3.0e4	15/15
0:cma	3* ³	5* ³	1* ³	2* ³	2* ²	15/15	0:cma	7* ³	6	1* ²	2* ²	2	15/15
1:f-auc	10	11	3	3	3	15/15	1:f-auc	25	11	2	2	3	15/15
f₁₄	10	58	139	251	476	15/15	f₁₄	75	304	932	1648	1.6e4	15/15
0:cma	2	4* ³	5* ³	5* ³	4* ³	15/15	0:cma	4* ³	4* ³	4* ³	6* ³	1* ³	15/15
1:f-auc	2	15	13	11	8	15/15	1:f-auc	33	38	23	19	3	15/15
f₁₅	511	1.9e4	2.0e4	2.1e4	2.1e4	14/15	f₁₅	3.0e4	3.1e5	3.2e5	4.5e5	4.6e5	15/15
0:cma	2	1	1	1	1	15/15	0:cma	1* ³	0.7* ³	0.7* ³	0.5* ³	0.5* ³	15/15
1:f-auc	5	7	7	7	7	12/15	1:f-auc	474	∞	∞	∞	∞	0/15
f₁₆	120	2662	1.0e4	1.2e4	1.2e4	15/15	f₁₆	1384	7.7e4	1.9e5	2.0e5	2.2e5	15/15
0:cma	3	2* ³	1.0* ³	0.9* ³	0.9* ³	15/15	0:cma	2* ³	0.9* ³	0.8* ³	1* ³	1* ³	15/15
1:f-auc	6	31	12	11	10	13/15	1:f-auc	∞	∞	∞	∞	∞	0/15
f₁₇	5	899	3669	6351	7934	15/15	f₁₇	63	4005	3.1e4	5.6e4	8.0e4	15/15
0:cma	5	1.0*	0.8*	0.8	1	15/15	0:cma	2* ²	1* ³	0.8* ³	1.0* ³	1*	15/15
1:f-auc	6	2	1	1	1	15/15	1:f-auc	23	7	2	2	5	13/15
f₁₈	103	3968	9280	1.1e4	1.2e4	15/15	f₁₈	621	2.0e4	6.8e4	1.3e5	1.5e5	15/15
0:cma	1* ³	0.9	1	1	1.0	15/15	0:cma	1* ³	1	1.0	1	1	15/15
1:f-auc	4	0.8	0.7	0.9*	1.0	15/15	1:f-auc	11	2	3	11	28	5/15
f₁₉	1	242	1.2e5	1.2e5	1.2e5	15/15	f₁₉	1	3.4e5	6.2e6	6.7e6	6.7e6	15/15
0:cma	21	125* ³	1* ³	1* ³	1* ³	15/15	0:cma	161* ³	0.7* ³	0.4* ³	0.4* ³	0.4* ³	15/15
1:f-auc	29	1726	11	11	10	5/15	1:f-auc	1315	∞	∞	∞	∞	0/15
f₂₀	16	3.8e4	5.4e4	5.5e4	5.5e4	14/15	f₂₀	82	3.1e6	5.5e6	5.6e6	5.6e6	14/15
0:cma	4	1	1	1	1	15/15	0:cma	5* ³	0.6* ³	0.6* ³	0.6* ³	0.6* ³	15/15
1:f-auc	7	7	5	5	5	10/15	1:f-auc	40	∞	∞	∞	∞	0/15
f₂₁	41	1674	1705	1729	1757	14/15	f₂₁	561	1.4e4	1.5e4	1.6e4	1.8e4	15/15
0:cma	6	30	31	31	31	14/15	0:cma	4* ²	110	106	100	88	7/15
1:f-auc	5	110	109	107	106	11/15	1:f-auc	12	568	547	515	456	3/15
f₂₂	71	938	1008	1040	1068	14/15	f₂₂	467	2.3e4	2.5e4	2.7e4	1.3e5	12/15
0:cma	12	166	161	158	155	11/15	0:cma	445	∞	∞	∞	∞	0/15
1:f-auc	5	803	747	724	706	6/15	1:f-auc	675	∞	∞	∞	∞	0/15
f₂₃	3	1.4e4	3.2e4	3.3e4	3.4e4	15/15	f₂₃	3	6.7e4	4.9e5	8.1e5	8.4e5	15/15
0:cma	2	33	15	14	14	11/15	0:cma	4	∞	∞	∞	∞	0/15
1:f-auc	2	2	3	5	7	15/15	1:f-auc	2	440	∞	∞	∞	0/15
f₂₄	1622	6.4e6	9.6e6	1.3e7	1.3e7	3/15	f₂₄	1.3e6	5.2e7	5.2e7	5.2e7	5.2e7	3/15
0:cma	3	1	0.9	0.7	0.7	2/15	0:cma	∞	∞	∞	∞	∞	0/15
1:f-auc	4	0.1	0.2	0.1	0.1	4/15	1:f-auc	∞	∞	∞	∞	∞	0/15

Table 12: Expected running time (ERT in number of function evaluations) divided by the best ERT measured during BBOB-2009 (given in the respective first row) for different Δf values for functions f_1 - f_{24} . The median number of conducted function evaluations is additionally given in *italics*, if $\text{ERT}(10^{-7}) = \infty$. 0:cma is IPOP-CMA-ES and 1:f-auc is F-AUC. Bold entries are statistically significantly better compared to the other algorithm, with $p = 0.05$ or $p = 10^{-k}$ where $k > 1$ is the number following the \star symbol, with Bonferroni correction of 48.

5.13 Summary Figures

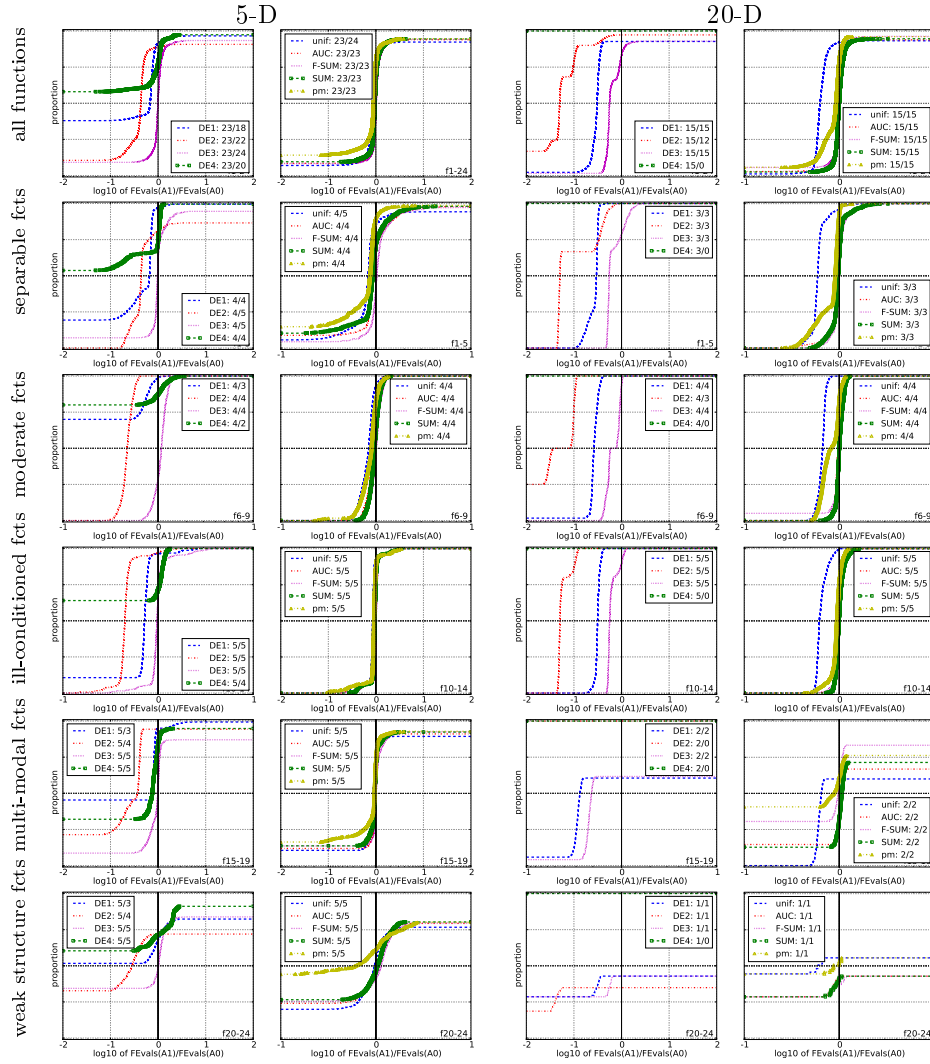


Figure 14: Empirical cumulative distributions (ECDF) of speed-up ratios in 5-D (left) and 20-D (right). ECDF of all trial pairs for each function of FEval ratios of F-AUC divided by ALG0 where ALG0 is successively DE/rand/1, DE/rand/2, DE/rand-to-best/2, DE/current-to-rand/1 (left-subcolumns) and successively uniform DE, AUC-Bandit, F-SR-Bandit, SR-Bandit, PM-AdapSS-DE (right-subcolumns). 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 (F-AUC first).

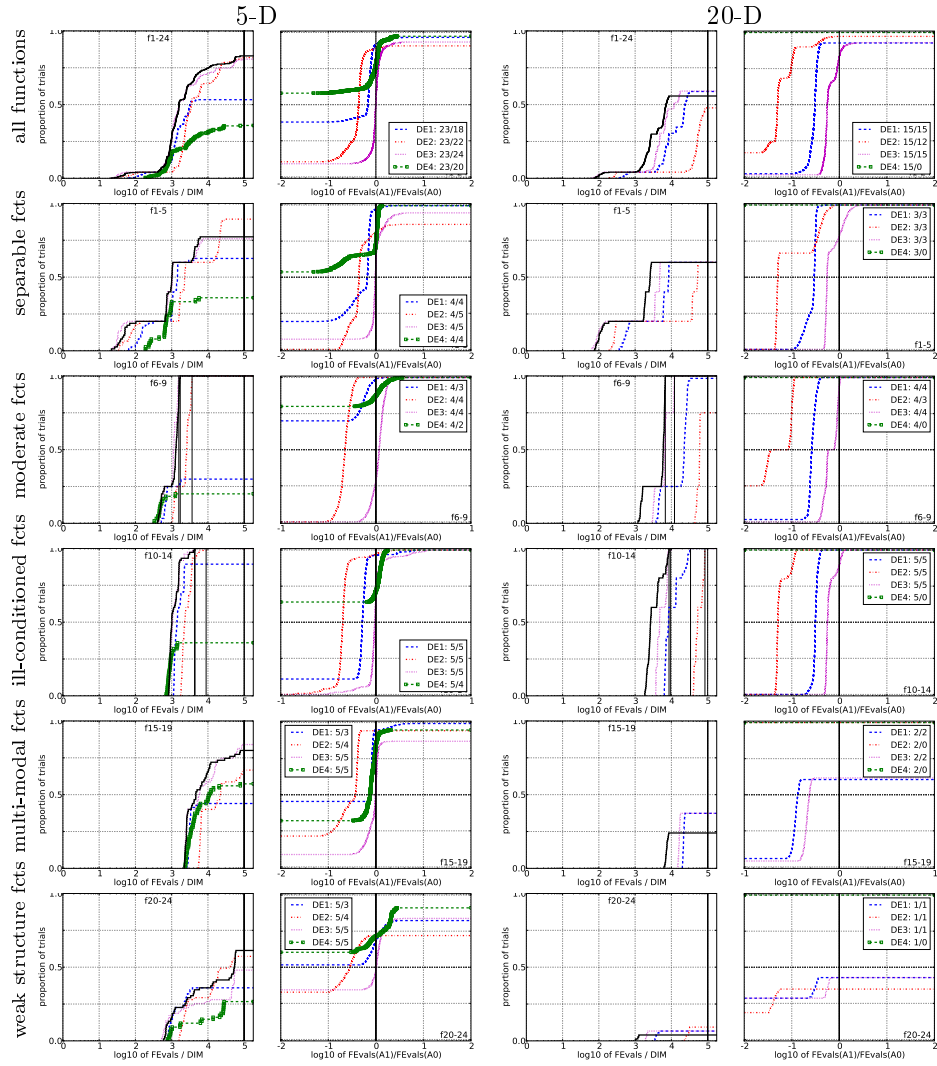


Figure 15: Empirical cumulative distributions (ECDF) of run lengths and speed-up ratios in 5-D (left) and 20-D (right).

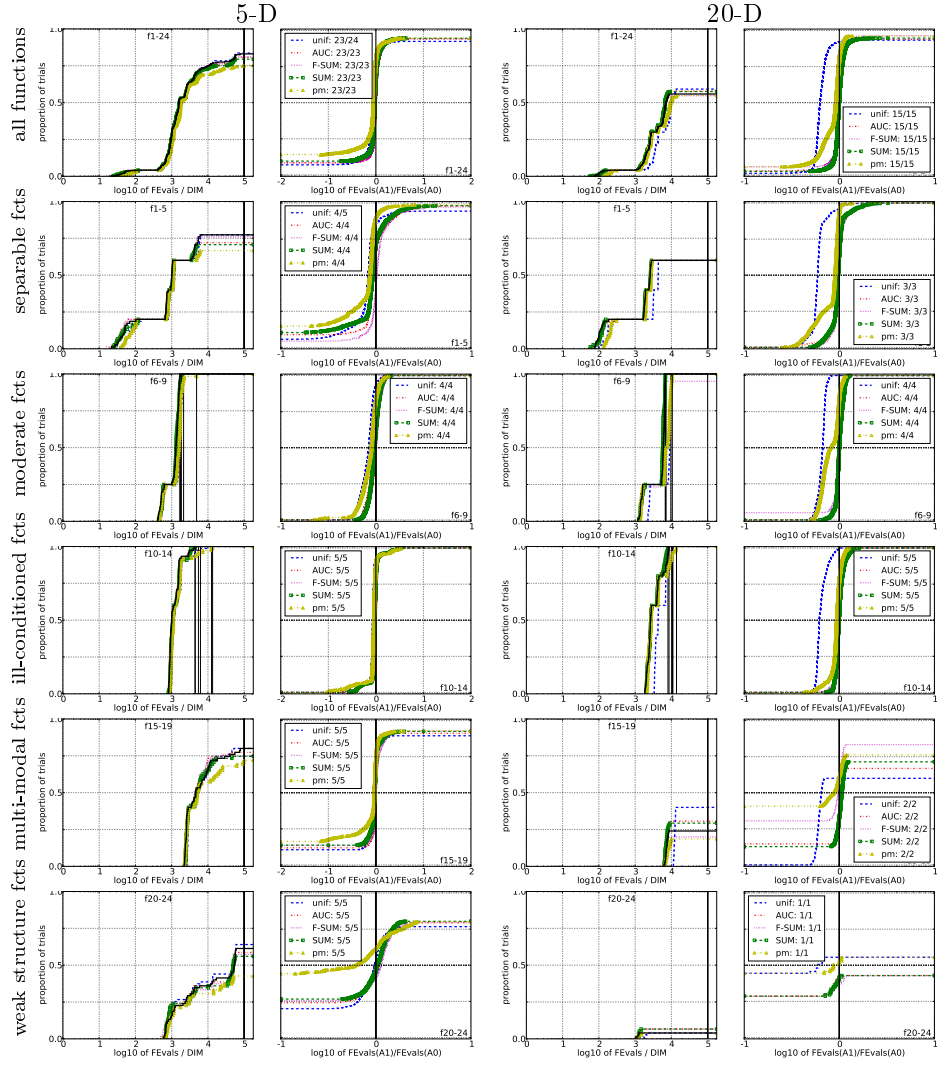


Figure 16: Empirical cumulative distributions (ECDF) of run lengths and speed-up ratios in 5-D (left) and 20-D (right).

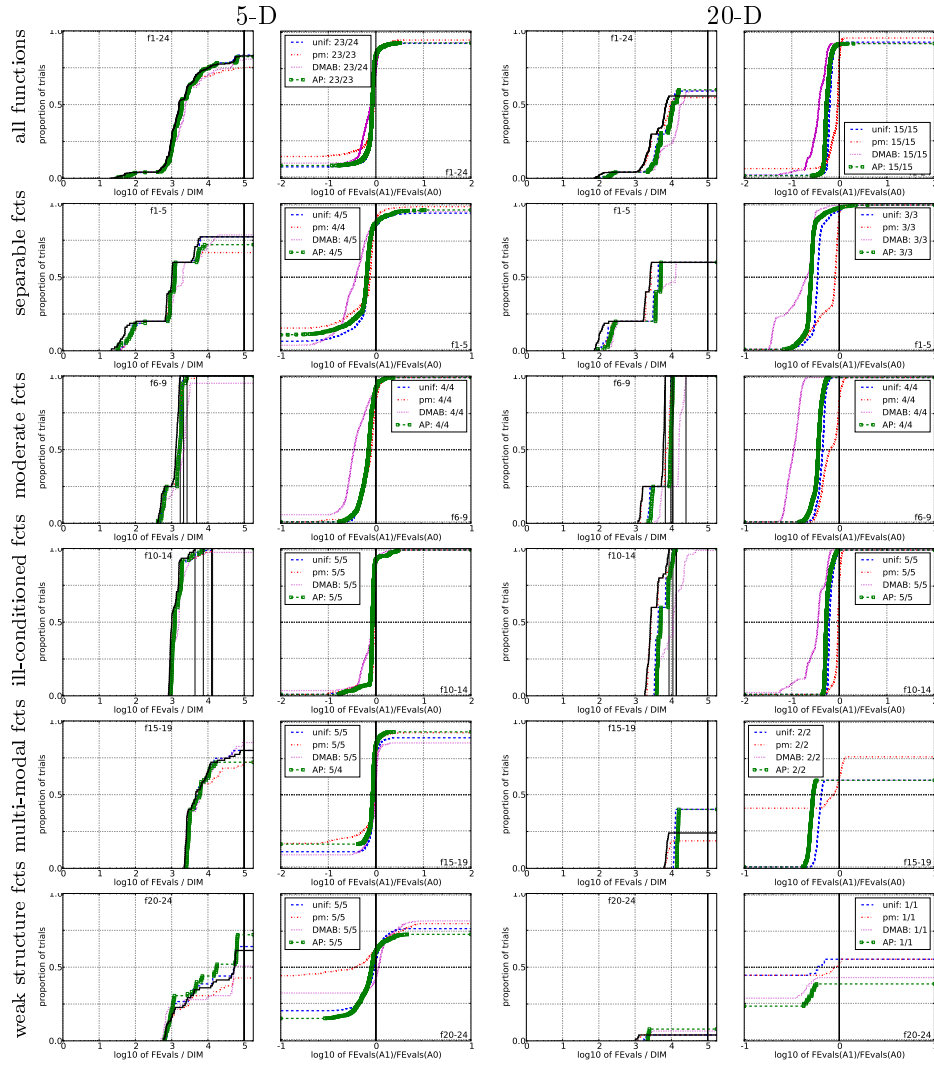


Figure 17: Empirical cumulative distributions (ECDF) of run lengths and speed-up ratios in 5-D (left) and 20-D (right).

Algo	2	1e1	1e2	1e3	1e4	1e5
F-AUC	1.1(0.58)	1 (0.06)	1.1(0.21)	1.0(0.16)	1 (0.40)	1 (0.00)
AUC	1.3(0.67)	1.2(0.51)	1.2(0.24)	1.1(0.22)	1.1 (0.11)	1.1 (0.30)
F-SUM	1.4(0.97)	1.2(0.42)	1.0(0.21)	1 (0.16)	0.99(0.40)	0.99(0.20)
SUM	1.5(0.73)	1.3(0.50)	1.2(0.23)	1.1(0.29)	1.1 (0.12)	1.1 (0.51)
unif	1.3(0.98)	1.2(0.62)	1.3(0.37)	1.2(0.42)	1.2 (0.28)	1.2 (0.28)
DE1	1.4(0.65)	1.4(0.64)	1.4(0.32)	1.4 (15)	3.5 (44)	2.4 (787)
DE2	1.3(0.59)	1.3 (1.5)	2.3(0.77)	2.3 (2.0)	2.3 (1.9)	2.2 (1.1)
DE3	1.4(0.57)	1.0(0.48)	1.0(0.32)	1.1(0.25)	1.0 (1.8)	1.0 (0.53)
DE4	1.3(0.82)	1.2(0.65)	1.0(0.23)	47 (204)	62 (198)	62 (651)
pm	1.1(0.29)	1.0(0.50)	1.2(0.29)	1.1(0.40)	1.3 (2.3)	1.2 (0.90)
DMAB	1.4(0.79)	1.2(0.88)	1.4(0.49)	1.5 (1.9)	1.7 (3.0)	1.5 (2.0)
AP	1.4(0.77)	1.2(0.43)	1.3(0.27)	1.2(0.39)	1.2 (0.54)	1.2 (0.32)

Table 13: Median ERT speed-up table over functions from f_1 to f_{24} in 5-D (interquartile range in brackets) for a number of functions FEvals. The ERT speed-up is computed as the ratio of the ERT of the algorithms considered (row) over the ERT of the algorithm which attained the smallest function value for the given budget FEvals.

Algo	2	1e1	1e2	1e3	1e4
F-AUC	1 (0.00)	1 (0.07)	1 (0.00)	1 (0.00)	1 (0.35)
AUC	1.0 (0.35)	1.0 (0.75)	1.3(0.30)	1.1(0.08)	1.0(0.65)
F-SUM	1 (0.46)	1.1 (0.23)	1.0(0.17)	1.0(0.07)	1.0 (2.0)
SUM	1 (0.51)	1.0 (0.57)	1.1(0.34)	1.1(0.08)	1.0(0.27)
unif	1.0 (0.40)	1.0 (0.68)	1.6(0.58)	1.7(0.27)	1.6(0.60)
DE1	0.97(0.52)	1.4 (1.4)	3.5 (1.4)	3.3(0.64)	3.2 (1.9)
DE2	0.92(0.49)	1.9 (4.8)	19 (11)	24 (11)	20 (195)
DE3	1.0 (0.49)	0.95(0.61)	1.7(0.61)	1.9(0.50)	1.8 (1.0)
DE4	1.1 (0.72)	1.1 (0.54)	1 (0.13)	28 (∞)	∞ (∞)
pm	1.0 (0.30)	1.2 (0.57)	1.3(0.32)	1.1(0.11)	1.1(0.63)
DMAB	1.1 (0.66)	1.4 (2.0)	10 (6.0)	7.9 (7.1)	3.5 (2.8)
AP	1 (0.39)	1.0 (0.79)	2.0(0.67)	2.0(0.27)	1.9(0.46)

Table 14: Median ERT speed-up table over functions from f_1 to f_{24} in 20-D (interquartile range in brackets) for a number of functions FEvals. The ERT speed-up is computed as the ratio of the ERT of the algorithms considered (row) over the ERT of the algorithm which attained the smallest function value for the given budget FEvals.

5.14 Summary Tables

References

- [1] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multi-armed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.
- [2] A. Auger and N. Hansen. A restart CMA evolution strategy with increasing population size. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2005)*, pages 1769–1776. IEEE Press, 2005.
- [3] M. Birattari, T. Stützle, L. Paquete, and K. Varrentrapp. A racing algorithm for configuring metaheuristics. In W. B. Langdon et al., editor, *Proc. GECCO*, pages 11–18. Morgan Kaufmann, 2002.
- [4] A. P. Bradley. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30:1145–1159, 1997.
- [5] A. Fialho, M. Schoenauer, and M. Sebag. Toward comparison-based adaptive operator selection. In J. Branke et al., editor, *Proc. Genetic Evol. Comput. Conf.* ACM Press, 2010.
- [6] 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.
- [7] D. E. Goldberg. Probability matching, the magnitude of reinforcement, and classifier system bidding. *Mach. Learn.*, 5(4):407–425, 1990.
- [8] W. Gong, A. Fialho, and Z. Cai. Adaptive strategy selection in differential evolution. In J. Branke et al., editor, *Proc. Genetic Evol. Comput. Conf.* ACM Press, 2010.
- [9] N. Hansen. Benchmarking a bi-population CMA-ES on the BBOB-2009 function testbed. In F. Rothlauf, editor, *GECCO (Companion)*, pages 2389–2396. ACM, 2009.
- [10] N. Hansen, A. Auger, S. Finck, and R. Ros. Real-parameter black-box optimization benchmarking 2010: Experimental setup. Technical Report RR-7215, INRIA, 2010.
- [11] 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.
- [12] A. W. Iorio and X. Li. Improving the performance and scalability of differential evolution. In *Proc. Conf. Simulated Evol. and Learning*, pages 131–140. Springer-Verlag, 2008.
- [13] K. Price. Differential evolution vs. the functions of the second ICEO. In *Proc. IEEE Congress on Evol. Comp.*, pages 153–157, 1997.
- [14] K. Price, R. Storn, and J. Lampinen. *Differential Evolution: A Practical Approach to Global Optimization*. Springer-Verlag, Berlin, 2005.

- [15] R. Storn and K. Price. Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Optim.*, 11(4):341–359, Dec 1997.
- [16] D. Thierens. An adaptive pursuit strategy for allocating operator probabilities. In *Proc. Genetic Evol. Comput. Conf.*, pages 1539–1546, 2005.



Centre de recherche INRIA Saclay – Île-de-France
Parc Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 Orsay Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399