Université Libre de Bruxelles

Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle

Computational results for an automatically tuned IPOP-CMA-ES on the CEC'05 benchmark set

Tianjun Liao, Marco A. Montes de Oca and Thomas Stützle

IRIDIA – Technical Report Series

Technical Report No.
TR/IRIDIA/2011-022

October 2011

IRIDIA – Technical Report Series
ISSN 1781-3794

Published by:

IRIDIA, *Institut de Recherches Interdisciplinaires et de Développements en Intelligence Artificielle*
Université Libre de Bruxelles
Av F. D. Roosevelt 50, CP 194/6
1050 Bruxelles, Belgium

Technical report number TR/IRIDIA/2011-022

Computational results for an automatically tuned IPOP-CMA-ES on the CEC'05 benchmark set

Tianjun Liao[1], Marco A. Montes de Oca[2], and Thomas Stützle[1]

[1] IRIDIA, CoDE, Université Libre de Bruxelles, Brussels, Belgium
{tliao,stuetzle}@ulb.ac.be
[2] Dept. of Mathematical Sciences, University of Delaware, Newark, DE, USA
mmontes@math.udel.edu

Abstract. In this article, we apply an automatic algorithm configuration tool to improve IPOP-CMA-ES's performance on the CEC'05 benchmark set. In particular, we consider a separation between tuning and test sets and, thus, tune IPOP-CMA-ES on a different set of functions than the ones of the CEC'05 benchmark set. Our experimental results show that the tuned IPOP-CMA-ES improves significantly over the default version of IPOP-CMA-ES. Furthermore, we show that the fine-tuned IPOP-CMA-ES is very competitive to state-of-the-art algorithms that use CMA-ES as a subordinate local search.

1 Introduction

The special session on real parameter optimization of the 2005 IEEE Congress on Evolutionary Computation (CEC'05) initiated a series of research efforts on benchmarking continuous optimizers and the development of new, improved continuous optimization algorithms. Two noteworthy results of this session are the establishment of a benchmark set of 25 hard, freely sizable benchmark functions and the establishment of IPOP-CMA-ES as the state-of-the-art continuous optimizer at least for what concerns the field of nature-inspired computation in the widest sense. In recent years, significant effort has been devoted by various researchers to improve the performance of IPOP-CMA-ES. Various authors have followed the path of embedding CMA-ES (without increasing population size) into other algorithms, using CMA-ES essentially as a local search. Few of these researches also have reported positive results, claiming better performance than IPOP-CMA-ES on the CEC'05 benchmark set [1, 2].

Here, we explore whether we can directly improve IPOP-CMA-ES's performance on the CEC'05 benchmark set by further fine-tuning IPOP-CMA-ES using algorithm configuration tools. In fact, IPOP-CMA-ES has a number of parameters and hidden constants in its code that make it a parameterized algorithm. Although its designers have spent a considerable effort in the design choices and certainly also in the definition of its parameters, over the last few years evidence has arisen that many algorithms' performance can be improved by considering automatic algorithm configuration and tuning tools [3–10].

It is therefore a natural question to ask whether and by how much the performance of IPOP-CMA-ES could be further improved by automatic algorithm configuration tools. Note that the answer to this question has also implications on the methodological aspects in algorithm development. When trying to improve over an algorithm such as IPOP-CMA-ES on a benchmark set, the design and tuning process of a new algorithm often starts by some new idea that is then iteratively refined manually until better performance on the benchmark is obtained. While this approach may have its general criticisms, in the case of IPOP-CMA-ES possible it is particularly critical since at least two approaches actually embed CMA-ES as a local search into another algorithm [1, 2]. An immediate question when observing the published results would be how IPOP-CMA-ES would perform against these "improved" algorithms if additional effort is put directly in IPOP-CMA-ES. In this article, we try to shed some light on this issue.

We have to remark that the present paper is not the first to try to further tune the performance of CMA-ES using automatic algorithm configuration tools. CMA-ES was used in the paper by Hutter et al. [7] as a benchmark algorithm to be tuned the authors considered to evaluate SPO^{+}, their improved variant of SPO [6]. However, they considered the tuning of CMA-ES restricted to only individual functions. Considering the tuned CMA-ES, they in this sense show that by "overtuning" CMA-ES on individual functions significantly improved performance over the default version of CMA-ES for this individual function could be obtained.[3] Another attempt of tuning CMA-ES was made by Smit and Eiben in the paper with the lurid title "Beating the World Champion Evolutionary Algorithm via REVAC Tuning" [11]. Unfortunately, they were not really using the "world champion evolutionary algorithm" in their study but a reduced version of it that has limitations on rotated functions (for details see Section 2). Nevertheless, they reported significant improvements of their tuned separable CMA-ES algorithm over the separable CMA-ES with default settings across the full range of functions contained in the CEC'05 benchmark set. From a tuning perspective, it is should here be mentioned that they were tuning their algorithm on the whole set of the CEC'05 benchmark functions. For the tuning, they allowed the separable CMA-ES on each 10 dimensional function a maximum of 100 000 function evaluations, but they were running as test runs the separable CMA-ES on the same functions for 1 000 000 function evaluations.

In this article, we tune IPOP-CMA-ES on a set of functions that has no overlap with the functions of the CEC'05 benchmark set. In this sense, we try to avoid a bias of the results obtained due to potentially overtuning the algorithm on the same benchmark functions as those on which the algorithm is tested. As such, we believe this gives a better assessment of the potential for what concerns the

[3] It is maybe relevant to remark here that the authors of [7] were simply continuing a usual practice of evaluating the tuning method SPO in their research on the further development of the SPO method.

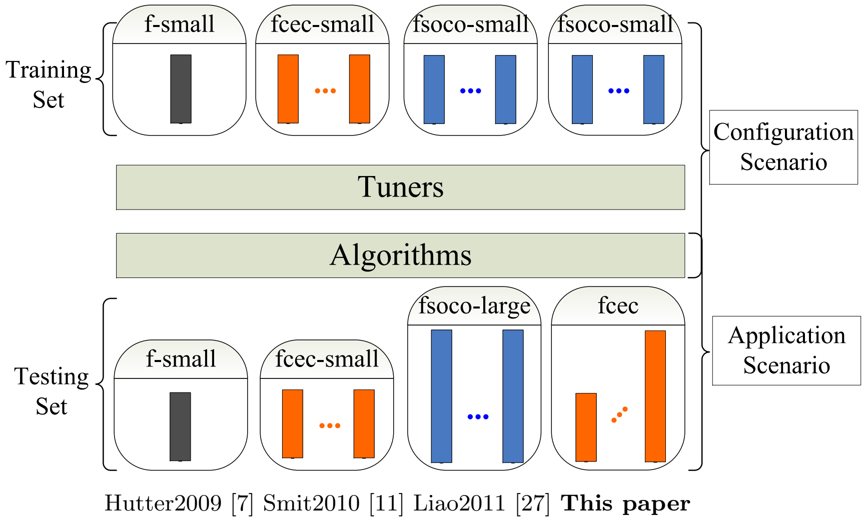

Fig. 1. Summary of the methodological approach to tuning CMA-ES over few recent articles. The approaches differ in the usage of a single versus multiple functions and the degree of separation between tuning and test set.

tuning of continuous optimizers as we have a separation between tuning and test set. Note, that such a separation is standard when studying tuning algorithms for combinatorial problems [4, 9, 12]. This separation of tuning and test sets for continuous functions as we apply it here is also different from our previous applications, where we have tuned algorithms on small dimensional functions and later tested them on much larger dimensional variants of the same functions. Note that in this case the training and testing functions are homogeneous even if their dimensionality is different. The differences in the applied approaches to tuning CMA-ES w.r.t to the separation of tuning and test set of functions is summarized in Figure 1.

As the tuning set, we consider small dimensional benchmark functions from the recent special issue of the Soft Computing journal [13, 14] on large-scale function optimization. This SOCO benchmark set contains 19 functions whose dimension is freely tunable. Four of these functions are the same as in the CEC'05 benchmark set, so we removed them from the tuning set. As tuner, we apply the irace software [15] to automatically tune seven parameters of IPOP-CMA-ES on the 10 dimensional SOCO benchmark functions (we refer to this tuned version of IPOP-CMA-ES as IPOP-CMA-ES-tsoco). Then we benchmark IPOP-CMA-ES-tsoco on the whole CEC'05 benchmark function suite for 10, 30 and 50 dimensions. The experimental results show that IPOP-CMA-ES-tsoco improves over the default parameter setting of IPOP-CMA-ES (called IPOP-CMA-ES-dp), and, maybe surprisingly, also improves over a version of IPOP-CMA-ES that we have tuned on the 10 dimensional CEC'05 benchmark set (we refer to this tuned version of IPOP-CMA-ES as IPOP-CMA-ES-tcec). Finally, we compare

IPOP-CMA-ES-tsoco with MA-LSch-CMA [1] and PS-CMA-ES [2], two state-of-the-art algorithms based on CMA-ES on CEC'05 benchmark function suite.

The article is structured as follows. In Section 2, we give some details on IPOP-CMA-ES and the parameters we considered for tuning. Section 3 gives details on the experimental setup and the automatic algorithm configuration tool that is used. The comparison of tuned IPOP-CMA-ES-tsoco to IPOP-CMA-ES-dp is given in Section 4.1; the comparison of tuned IPOP-CMA-ES-tsoco to IPOP-CMA-ES-tcec in Section 4.2; the comparison to other state-of-the-art algorithms in Section 4.3. We end by a discussion of related results in Section 5 and we conclude in Section 6.

2 Parameterized IPOP-CMA-ES

CMA-ES [16–18] is an (μ, λ)-evolution strategy that samples new candidate solutions based on a multivariate normal distribution that is adapted at execution time. In particular, CMA-ES adapts the full covariance matrix of a normal search distribution. It is shown to result in a search that is invariant against linear transformations of the search space (rotational invariance), which makes it particularly suited for rotated functions. Sep-CMA-ES [19, 20] is a modification of CMA-ES with lower time complexity that instead of the full covariance matrix uses a diagonal matrix (that is, the covariances are assumed to be zero); in a sense, in Sep-CMA-ES the step size for each variable is adapted independently of the other variables. Sep-CMA-ES is also the variant that was used in the paper by Smit and Eiben [11], which was mentioned in the introduction. IPOP-CMA-ES is a variant of the CMA-ES algorithm that uses a restart schema coupled with an increasing population size. For a detailed explanation of the optimization principles of CMA-ES and IPOP-CMA-ES we refer to [21, 22].

The default settings of IPOP-CMA-ES are as follows. The initial population size is $\lambda = 4 + \lfloor 3 \ln(D) \rfloor$, where D is the number of dimensions of the function to be optimized. The number of selected search points in the parent population is $\mu = \lfloor 0.5\lambda \rfloor$. The initial step-size is $\sigma^{(0)} = 0.5(B - A)$, where $[A, B]^n$ is the initial search interval. At each restart, the population size is multiplied by a factor of two. Restarts occur if the stopping criterion is met. The three parameters *stopTolFunHist*, *stopTolFun* and *stopTolX* of the stopping criterion refer to the range of the improvement of the best objective function values in the last generations, all function values of the recent generation, and the standard deviation of the normal distribution, respectively. These internal parameter settings of IPOP-CMA-ES are set, as far as we are aware, by the experience of the developers of IPOP-CMA-ES. For detailed definitions of the stopping criteria but also for an explanation of all the, in part, complex details of CMA-ES, we refer to [22].

For tuning IPOP-CMA-ES, we considered seven parameters related to the above mentioned default settings. The parameters are given in Table 1. The first four parameters are actually used in a formula to compute some internal parameter of IPOP-CMA-ES and the remaining three to define its termination.

Table 1. Parameters that have been considered for tuning. Given are the default values of the parameters and the continuous range we considered for tuning. The last two columns give for each set of tuning instances the found algorithm configurations.

Parameters	Formulas	Factor	Def	Range	Tuning instances	
					$f_{cec}*$	$f_{soco}*$
Pop size (λ)	$4 + \lfloor a \ln(D) \rfloor$	a	3	[1, 10]	7.315	9.600
Parent size (μ)	$\lfloor \lambda / b \rfloor$	b	2	[1, 5]	3.776	1.452
Init step size ($\sigma^{(0)}$)	$c(B - A)$	c	0.5	(0,1)	0.8297	0.6034
IPOP factor(d)	d	d	2	[1, 4]	2.030	3.292
stopTolFun	10^e	e	-12	[-20, -6]	-8.104	-8.854
stopTolFunHist	10^f	f	-20	[-20, -6]	-6.688	-9.683
stopTolX	10^g	g	-12	[-20, -6]	-13.85	-12.55

Note that if a run of IPOP-CMA-ES is terminated, CMA-ES is restarted with an increased population size λ. For the increase of the population size, we here introduce a parameter d we call IPOP factor. The first five columns of Table 1 give the parameters we use, the formula where they are used, their default values and the range that we considered for tuning. The remaining two columns are explained later.

3 Experimental Setup

We used the C version of IPOP-CMA-ES (last modification date 10/10/16) from Hansen's webpage `http://www.lri.fr/~hansen/cmaesintro.html`. We modified the code to handle bound constraints by clamping the variable values outside the bounds on the nearest bound value. (The issues about the effects of enforcing and ignoring bound constraints have been addressed by Liao et al. [23]) Our test-suite consists of 25 CEC'05 benchmark functions (functions labeled as $f_{cec}*$) of dimensions $n \in \{10, 30, 50\}$. The training instances of IPOP-CMA-ES-tsoco and IPOP-CMA-ES-tcec involve the 10-dimensional SOCO and CEC'05 benchmark functions, respectively. Unfortunately, the SOCO and CEC'05 benchmark sets overlap on four functions and therefore we have removed these four functions from the SOCO benchmark set that we used for tuning. In particular, we have eliminated the four SOCO functions f_{soco1}, f_{soco3}, f_{soco4} and f_{soco8}, which are (except for the shift vectors for moving the known optimum solution and the domain range) the same as the CEC'05 functions f_{cec1}, f_{cec6}, f_{cec9} and f_{cec2}, respectively.

The properties of the CEC'05 and SOCO benchmark functions are listed in Table 2; for a detailed description, we refer to [13, 24]. We followed the protocol described in [24] for the CEC'05 test-suite, that is, the maximum number of function evaluations was $10\,000 \times D$ where $D \in \{10, 30, 50\}$ is the dimensionality of a function. The investigated algorithms were run 25 times on each function. We report error values defined as $f(\boldsymbol{x}) - f(\boldsymbol{x}^*)$, where $\boldsymbol{x}$ is a candidate solution and $\boldsymbol{x}^*$ is the optimal solution. Error values lower than 10^{-8} are clamped to

10^{-8}, which is our zero threshold. Our analysis considers the median errors, average errors and the solution quality distribution for each function.

For tuning the parameters of IPOP-CMA-ES, we employ Iterated F-Race [25], a racing algorithm for algorithm configuration that is included in the irace package [15]. Iterated F-Race is an algorithm that repeatedly applies F-Race to a set of candidate configurations that are generated via a rather simplistic sampling mechanism that intensifies the search around the best found configurations. The generated candidate configurations then perform a "race", where each candidate configuration is iteratively applied to an instance of a combinatorial problem or a function in the continuous optimization case. Poor performing candidate configurations are eliminated from the race based on the result of statistical tests. In particular, the Friedman test by ranks checks whether sufficient statistical evidence is gathered that indicates that some configurations behave differently from the rest. If the null hypothesis of the F-test is rejected, Friedman post-tests are used to eliminate the statistically worse performing candidates.

When applied to tuning continuous functions, a useful feature of F-Race is ranking. Ranking provides a straightforward way of blocking and, in this way, to account for the different ranges of the values of the benchmark functions. Without ranking, few functions with large values would dominate the evaluation of the algorithm performance. This is particularly relevant for continuous optimization problems due to the widely different difficulties of the benchmark functions and the widely different evaluation value ranges.

The performance measure for tuning is the fitness error value of each instance. In the automatic parameter tuning process, the maximum budget is set to 5 000 runs of IPOP-CMA-ES. The setting of Iterated F-Race we used are the default [15]. When tuning parameters on SOCO training instances, 10 dimensional f_{soco1}-f_{soco19} (except f_{soco1}, f_{soco3}, f_{soco4} and f_{soco8}) were sampled as training instances in a random order and the number of function evaluations of each run is set equal to $5\,000 \times D$ ($D = 10$). When tuning parameters on CEC'05 training instances, the 10 dimensional variants of f_{cec1}-f_{cec25} were sampled as training instances in a random order and the number of function evaluations of each run is equal to $10\,000 \times D$ ($D = 10$). The default and tuned settings of IPOP-CMA-ES' parameters are presented in Table 1.

The results of the different algorithmic variants are compared from a statistical perspective in two ways. First, on an instance level, we use a two-sided Wilcoxon signed-rank test at the 0.05 α-level to check whether the performance of two algorithms is statistically significantly different. Recall that each algorithm is run 25 independent times on each benchmark function. Second, across all benchmark functions, we apply a two-sided Wilcoxon matched-pairs signed-rank test at the 0.05 α-level to check whether the differences in the mean or median results obtained by two algorithms on each of the 25 CEC'05 benchmark functions is statistically significant.

4 Experimental study

4.1 IPOP-CMA-ES-tsoco vs. IPOP-CMA-ES-dp

Table 3 shows the performance of IPOP-CMA-ES-dp (default parameters) and IPOP-CMA-ES-tsoco (tuned on SOCO benchmark functions) on the CEC'05 benchmark function suite. Considering the differences on individual functions, we first observe that IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco reach on surprisingly many functions similar performance at least from the perspective of the applied Wilcoxon test: on 19, 17, and 14 functions for 10, 30, and 50 dimensions, respectively, no statistically significant differences could be observed. On one side, this may be due to the relatively small number of 25 independent runs of the algorithms; on the other side, this is also caused by the fact that several benchmark functions are very easy to solve and therefore introduce floor effects. For example, functions f_{cec1}, f_{cec2}, f_{cec3}, f_{cec6}, and f_{cec7} are solved by IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco in all runs and in all dimensions to the zero threshold. In the other functions, IPOP-CMA-ES-tsoco performs better than IPOP-CMA-ES-dp except in 2, 1, and 1 cases for dimensions 10, 30, and 50, respectively, indicating superior performance of IPOP-CMA-ES-tsoco over IPOP-CMA-ES-dp.

When focusing on the comparison of the means and medians obtained across each of the benchmark functions, we can observe that IPOP-CMA-ES-tsoco reaches statistically better results on dimension 50 according to the Wilcoxon test, while on dimensions 10 and 30 the observed differences are not statistically significant. However, it should be reminded that on a function-by-function basis IPOP-CMA-ES-tsoco is statistically better than IPOP-CMA-ES-dp on more functions than vice-versa.

On few functions, the differences in the solution qualities are very strong. As an example, consider function f_{cec4}, where IPOP-CMA-ES-dp stagnated at very high average error values of 6.58E+02 and 1.43E+04 for dimensions 30 and 50, respectively, while IPOP-CMA-ES-tsoco reached in each trial a solution better than the zero threshold. On other functions of dimension 50, such as functions f_{cec12}, f_{cec16}, and f_{cec17}, IPOP-CMA-ES-tsoco more than halved the error values that were reached by IPOP-CMA-ES-dp.

For avoiding a possible bias caused by the relatively small sample size 25 and increasing the power of the statistical test, we re-evaluate IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco using a sample size of 100. The resulting performance of IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco is shown in Table 4. IPOP-CMA-ES-tsoco keeps its superiority over IPOP-CMA-ES-dp. Moreover, IPOP-CMA-ES-tsoco reaches now statistically significantly better performance on the distribution of median error values according to the Wilcoxon test on dimension 30. The results confirm its superiority on dimension 50.

As a further visual illustration, Fig 2 shows correlation plots where each point has as x and y coordinate, the average errors obtained with IPOP-CMA-ES-tsoco and IPOP-CMA-ES-dp, respectively. Each plot of Fig 2 shows the average errors for 10, 30, and 50 dimensional problems, respectively. Besides giving a graphical illustration of the results reported previously in the tables, the plots also show clearly that on some functions, IPOP-CMA-ES-tsoco reaches results that are much better solution quality than those of IPOP-CMA-ES-dp (see the circles that are distant from the diagonal). This is the case especially on functions f_{cec4}, f_{cec5}, f_{cec11}, f_{cec12}, f_{cec16} and f_{cec17}. Next, we give a more detailed analysis of the algorithm behavior on these six functions.

Fig 3 shows the development of the average error for IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco over the number of function evaluations on functions f_{cec4}, f_{cec5}, f_{cec11}, f_{cec12}, f_{cec16} and f_{cec17} of dimension 50. We observe that IPOP-CMA-ES-tsoco and IPOP-CMA-ES-dp perform similar up to about 1.00E+04 function evaluations. As the number of function evaluation increases, the advantage of IPOP-CMA-ES-tsoco over IPOP-CMA-ES-dp starts to become apparent. At the stopping criterion of 5.00E+05 function evaluations from the CEC'05 competition rules (indicated by the dotted, vertical lines in the plots), IPOP-CMA-ES-tsoco shows generally much lower average errors. Especially on f_{cec4} and f_{cec5}, IPOP-CMA-ES-tsoco converges fast to the zero threshold after 1.00E+04 function evaluations. If we consider a larger number of function evaluations by considering the results obtained of up to 5.00E+06 function evaluations, we can see that IPOP-CMA-ES-dp catches up with the lower average errors of IPOP-CMA-ES-tsoco on functions f_{cec5}, f_{cec11}, and f_{cec16}. Hence, on these functions the result of IPOP-CMA-ES-tsoco parameter settings appears to result simply in a faster convergence towards near-optimal solutions. On functions f_{cec4} and f_{cec12} the advantage of IPOP-CMA-ES-tsoco w.r.t. the average error remains substantial.

Changing perspective, we next look at the performance on these six functions using run-length distributions (RLDs) [26]. In particular, we consider qualified RLDs, which give the distribution of the number of function evaluations to reach specific bounds on the errors, which for the six functions are taken to be 1.00E+03, 1.00E−08, 1.00E−08, 1.00E+05, 1.00E+00 and 1.00E+02 for the 50 dimensional versions of functions f_{cec4}, f_{cec5}, f_{cec11}, f_{cec12}, f_{cec16} and f_{cec17}, respectively. Fig 4 shows these qualified RLDs measured across 100 independent runs obtained by IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco. Both, IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco have a 100% success rate to reach the solution quality demanded on each of these functions. However, it is clear that the different settings of IPOP-CMA-ES give a very different impact on their convergence behavior. Clearly, IPOP-CMA-ES-tsoco reaches faster than IPOP-CMA-ES-dp a 100% success rate; on some of these functions (e.g. f_{cec4} and f_{cec5}) it is order of magnitudes faster. In fact, on four functions, f_{cec4}, f_{cec5}, f_{cec12} and f_{cec17}, IPOP-CMA-ES-tsoco completely dominates IPOP-CMA-ES-dp w.r.t. the RLDs.

Finally, we consider qualified RLDs for IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco on functions f_{cec1}, f_{cec2}, f_{cec3}, f_{cec6} and f_{cec7} on dimension 50. On these functions each trial of IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco reaches the zero threshold within the termination criterion of the CEC'05 protocol. Fig 5 shows the run-length distributions over 100 independent runs obtained by IPOP-

CMA-ES-dp and IPOP-CMA-ES-tsoco on these five functions. We observe that on f_{cec1}, f_{cec2} and f_{cec3}, both IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco converge very fast to zero threshold in each trial using a relatively small number of function evaluations without recurring to restarts. For these three, relative easy functions, IPOP-CMA-ES-tsoco converge slightly slower than IPOP-CMA-ES-dp mainly because of its larger initial population size. On f_{cec6} and f_{cec7}, in contrast, IPOP-CMA-ES-tsoco reaches a 100% success rate faster than IPOP-CMA-ES-dp, although there is no dominance relationships among the RLDs, differently from what we observed on some of the more difficult functions.

4.2 IPOP-CMA-ES-tsoco vs. IPOP-CMA-ES-tcec

Table 5 shows the performance of IPOP-CMA-ES-tcec and IPOP-CMA-ES-tsoco on the CEC'05 benchmark set. IPOP-CMA-ES-tcec and IPOP-CMA-ES-tsoco are mutually statistically better than each other on four functions of dimension 10, respectively. On 10 dimensional functions, IPOP-CMA-ES-tsoco is slightly worse than IPOP-CMA-ES-tcec w.r.t the distribution of mean or median errors. This may be a result of the fact that IPOP-CMA-ES-tcec is tuned on the 10 dimensional CEC'05 benchmark set. Interestingly, this slight superiority of IPOP-CMA-ES-tcec does not generalize to higher dimensions. As an example, consider functions f_{cec18}, f_{cec19}, f_{cec20} of dimension 10. On these, IPOP-CMA-ES-tcec obtains an error value of 3.00E+02 in all independent 25 runs which is the lowest value reported in the literature for these functions as far as we aware. However, on the 50 dimensional version of these functions, IPOP-CMA-ES-tcec is significantly worse than IPOP-CMA-ES-tsoco. Moreover, considering the differences on all functions of dimension 50, IPOP-CMA-ES-tsoco statistically significantly improves upon IPOP-CMA-ES-tcec on 12 functions while it performs statistically significantly worse than IPOP-CMA-ES-tcec on only 3 functions. Considering the distribution of the mean or median error values of the functions with 50 dimensions, IPOP-CMA-ES-tsoco statistically significantly improves upon IPOP-CMA-ES-tcec.

In addition to the superios performance of IPOP-CMA-ES-tsoco over IPOP-CMA-ES-tcec for higher dimensional functions, it should also be mentioned that tuning on the SOCO benchmark functions is much faster than on the CEC'05 benchmark set. In fact, the difference in computation time amounts to a factor of about 50. This difference is mainly due to the facts that (i) 64% of the CEC'05 functions of each dimension are rotated functions, which requires more costly computations in the evaluation (such as multiplication operations on a rotated matrix) and (ii) 44% of the CEC'05 functions of each dimension are composed functions.

4.3 Comparison to state-of-the-art methods that exploit CMA-ES

We compared IPOP-CMA-ES-tsoco with two state-of-the-art algorithms that use CMA-ES as an underlying local search method; these are a memetic algorithms with local search chains based on CMA-ES (MA-LSch-CMA) [1] and a hybridization of a PSO algorithm with CMA-ES (PS-CMA-ES) [2]. Our comparisons follows the experimental analysis used in [1] and [2], that is by (i) statistically analyzing for the distribution of the average errors as in [1] and, (ii) ranking the average errors as in [2]. For the results of MA-LSch-CMA and PS-CMA-ES we use those reported in [23]. Note that the original results reported in [1,2] had issues with the satisfaction of the bound constraints of the CEC'05 benchmark functions, which is not anymore the case with those results reported in [23]. Table 6 shows that IPOP-CMA-ES-tsoco performs statistically significantly better than MA-LSch-CMA in all dimensions; IPOP-CMA-ES-tsoco performs statistically significantly better than PS-CMA-ES on the 50 dimensional functions and it reaches better performance than PS-CMA-ES on more funcitons for dimensions 10 and 30. Table 6 also shows that IPOP-CMA-ES-tsoco obtains the best average ranking in all dimensions. IPOP-CMA-ES-tsoco also reaches most often the zero threshold in all dimensions.

5 Discussion

We also compare IPOP-CMA-ES-tsoco to Sep-IPOP-CMA-ES-tsoco [27], the algorithm used in [11], on the full CEC'05 benchmark set. Fig. 6 shows correlation plots that illustrate the relative performance for Sep-IPOP-CMA-ES-tsoco and IPOP-CMA-ES-tsoco on dimensions 10, 30 and 50 respectively. Each point represents the average error value obtained by either of the two algorithms. We observe that IPOP-CMA-ES-tsoco is statistically significantly better than Sep-IPOP-CMA-ES-tsoco on the distribution of mean error values on all dimensions. The data of Sep-IPOP-CMA-ES-tsoco on the full CEC'05 benchmark set are given in the supplementary page `http://iridia.ulb.ac.be/supp/IridiaSupp2011-022`. This comparison confirms our expectation of IPOP-CMA-ES-tsoco's superiority on CEC'05 benchmark set, which involves 64% rotated functions for each dimension. The most significant example is f_{cec3}, a unimodal rotated high conditional function, where Sep-IPOP-CMA-ES-tsoco stagnated at very high average error values for all dimensions, while IPOP-CMA-ES-tsoco reached in each trial the zero threshold. However, Sep-IPOP-CMA-ES-tsoco obtains a better performance than IPOP-CMA-ES-tsoco on f_{cec10}, a rotated Rastrigin function, over all dimensions. This case gives an indication that we can only conclude that IPOP-CMA-ES's rotational invariance plays a pivotal role to handle most but not all rotated functions.

Next, we take the reported average errors of the results reported for IPOP-CMA-ES in the CEC'05 special session as a reference and refer to these results as IPOP-CMA-ES-05. We summarize the comparison with IPOP-CMA-ES-tsoco in Table 7. IPOP-CMA-ES-tsoco performs statistically significantly better than IPOP-CMA-ES-05 on dimension 30 and it reaches on more functions statistically significantly better results (on a per function bases). This confirms the high performance of IPOP-CMA-ES-tsoco.

We also tuned IPOP-CMA-ES using variations of the some settings in the irace method and using different sets of tuning instances. In particular, we (i) replaced the F-test with a t-test (that is, using t-race); (ii) increased the tuning budget to 25000 runs; (iii) tuned across all 19 functions in SOCO, and (iv) tuned only across multi-modal functions in SOCO. Similar to IPOP-CMA-ES-tsoco, these trials improved upon IPOP-CMA-ES-dp. Especially on dimension 50, the tuned configurations returned by all these four trials are statistically significantly better than IPOP-CMA-ES-dp on the distribution of the mean or the median error values. The data of these trials are also given in the supplementary page `http://iridia.ulb.ac.be/supp/IridiaSupp2011-022`.

6 Conclusions and Future Work

In this article, we tuned IPOP-CMA-ES to improve its performance on the CEC'05 benchmark set. We do so by using a separation between tuning and test set to avoid the bias of the results due to potentially overtuning the algorithm. Our experimental results suggest that the tuned IPOP-CMA-ES improves over the default parameter settings of IPOP-CMA-ES-dp. IPOP-CMA-ES-tsoco also performs competitive or better than advanced methods (MA-LSch-CMA and PS-CMA-ES) designed to improve over IPOP-CMA-ES; furthermore, IPOP-CMA-ES-tsoco outperforms Sep-IPOP-CMA-ES-tsoco and gives also improvements over the results reported by the implementation of IPOP-CMA-ES that was used to obtain the results for the CEC'05 benchmark competition. Our ongoing work is to conduct a more thorough analysis and comparisons among different, automatically tuned continuous optimizers.

7 Acknowledgments

The authors would like to thank Dr. Nikolaus Hansen for the IPOP-CMA-ES codes. The authors also would like to thank Dr. Daniel Molina and Dr. Christian L. Müller for their updated results of MA-LSch-CMA and PS-CMA-ES on the CEC'05 benchmark function suite (taken from [23]). This work was supported by the Meta-X project funded by the Scientific Research Directorate of the French Community of Belgium. Thomas Stützle acknowledges support from the Belgian F.R.S.-FNRS, of which he is a Research Associate. Tianjun Liao acknowledges a fellowship from the China Scholarship Council.

References

1. Molina, D., Lozano, M., García-Martínez, C., Herrera, F.: Memetic algorithms for continuous optimisation based on local search chains. Evolutionary Computation **18**(1) (2010) 27–63
2. Muller, C., Baumgartner, B., Sbalzarini, I.: Particle swarm CMA evolution strategy for the optimization of multi-funnel landscapes. In: Proceeding of IEEE Congress on Evolutionary Computation, CEC 2009, Piscataway, NJ, IEEE Press (2009) 2685–2692

3. Adenso-Diaz, B., Laguna, M.: Fine-tuning of algorithms using fractional experimental designs and local search. Operations Research **54** (2006) 99–114
4. Birattari, M., Stützle, T., Paquete, L., Varrentrapp, K.: A racing algorithm for configuring metaheuristics. In: Proceedings of the Genetic and Evolutionary Computation Conference. GECCO'02, San Francisco, CA, USA, Morgan Kaufmann (2002) 11–18
5. Balaprakash, P., Birattari, M., Stützle, T.: Improvement strategies for the f-race algorithm: sampling design and iterative refinement. In Bartz-Beielstein, et al., eds.: Proceedings of the 4th International Conference on Hybrid Metaheuristics. Volume 4771 of LNCS. Springer, Berlin, Germany (2007) 108–122
6. Bartz-Beielstein, T.: Experimental Research in Evolutionary Computation: the New Experimentalism. Springer, Berlin, Germany (2006)
7. Hutter, F., Hoos, H.H., Leyton-Brown, K., Murphy, K.P.: An experimental investigation of model-based parameter optimisation: SPO and beyond. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO'09, New York, NY, USA, ACM (2009) 271–278
8. Hutter, F., Babic, D., Hoos, H., Hu, A.: Boosting verification by automatic tuning of decision procedures. In: Proceedings of Formal Methods in Computer Aided Design, FMCAD07, Piscataway, NJ, IEEE Press 27–34
9. Hutter, F., Hoos, H., Leyton-Brown, K., Stützle, T.: ParamILS: an automatic algorithm configuration framework. Journal of Artificial Intelligence Research **36**(1) (2009) 267–306
10. Nannen, V., Eiben, A.E.: Relevance estimation and value calibration of evolutionary algorithm parameters. In: Proceedings of the 20th International Joint Conference on Artifical Intelligence, San Francisco, CA, USA, Morgan Kaufmann (2007) 975–980
11. Smit, S., Eiben, A.: Beating the World Champion Evolutionary Algorithm via REVAC Tuning. In: Proceeding of IEEE Congress on Evolutionary Computation, CEC 2010, IEEE Press, Piscataway, NJ (2010) 1–8
12. Birattari, M.: Tuning Metaheuristics: A Machine Learning Perspective. 1st ed. 2005. 2nd printing edn. Springer, Berlin, Germany (2009)
13. Herrera, F., Lozano, M., Molina, D.: Test suite for the special issue of soft computing on scalability of evolutionary algorithms and other metaheuristics for large scale continuous optimization problems (2010) URL: `http://sci2s.ugr.es/eamhco/`.
14. Lozano, M., Molina, D., Herrera, F.: Editorial scalability of evolutionary algorithms and other metaheuristics for large-scale continuous optimization problems. Soft Computing - A Fusion of Foundations, Methodologies and Applications **15** (2011) 2085–2087
15. López-Ibáñez, M., Dubois-Lacoste, J., Stützle, T., Birattari, M.: The `irace` package, iterated race for automatic algorithm configuration. Technical Report TR/IRIDIA/2011-004, IRIDIA, Université Libre de Bruxelles, Belgium (2011)
16. Hansen, N., Ostermeier, A.: Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In: Proceedings of IEEE International Conference on Evolutionary Computation, CEC 1996, IEEE Press (1996) 312–317
17. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation **9**(2) (2001) 159–195
18. Hansen, N., Muller, S., Koumoutsakos, P.: Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evolutionary Computation **11**(1) (2003) 1–18

19. Ros, R., Hansen, N.: A simple modification in cma-es achieving linear time and space complexity. In Rudolph, G., Jansen, T., Lucas, S., Poloni, C., Beume, N., eds.: Parallel Problem Solving from Nature PPSN X. Volume 5199 of LNCS., Springer, Berlin, Germany (2008) 296–305
20. Ros, R.: Benchmarking sep-cma-es on the bbob-2009 noisy testbed. In: Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers. GECCO'09, New York, NY, USA, ACM (2009) 2441–2446
21. Hansen, N.: The CMA Evolution Strategy: A Tutorial. (2010) Online: http://www. lri. fr/hansen/cmatutorial. pdf.
22. Auger, A., Hansen, N.: A restart CMA evolution strategy with increasing population size. In: Proceeding of IEEE Congress on Evolutionary Computation, CEC 2005, Piscataway, NJ, IEEE Press (2005) 1769–1776
23. Liao, T., Molina D., Montes de Oca., M.A., Stützle, T.: A Note on the Effects of Enforcing Bound Constraints on Algorithm Comparisons using the IEEE CEC'05 Benchmark Function Suite. Technical Report TR/IRIDIA/2011-010, IRIDIA, Université Libre de Bruxelles, Belgium (2011)
24. Suganthan, P., Hansen, N., Liang, J., Deb, K., Chen, Y., Auger, A., Tiwari, S.: Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. Technical Report 2005005, Nanyang Technological University (2005)
25. Birattari, M., Yuan, Z., Balaprakash, P., Stützle, T.: F-Race and iterated F-Race: An overview. In Bartz-Beielstein, et al., eds.: Experimental Methods for the Analysis of Optimization Algorithms. Springer, Berlin, Germany (2010) 311–336
26. Hoos, H., Stützle, T.: Stochastic Local Search: Foundations & Applications. Morgan Kaufmann, San Francisco, CA, USA (2004)
27. Liao, T., Montes de Oca, M., Stützle, T.: Tuning Parameters across Mixed Dimensional Instances: A Performance Scalability Study of Sep-G-CMA-ES. In: Proceedings of the Workshop on Scaling Behaviours of Landscapes, Parameters and Algorithms of the Genetic and Evolutionary Computation Conference, GECCO'11, New York, NY, USA, ACM (2011) 703–706

Table 2. Benchmark functions

ID	Name/Description	Range $[X_{\min}, X_{\max}]^D$	Uni/Multi modal	Sepa-rable	Rotat-ed
f_{cec1}	Shift.Sphere	$[-100,100]^D$	U	Y	N
f_{cec2}	Shift.Schwefel 1.2	$[-100,100]^D$	U	N	N
f_{cec3}	Shift.Ro.Elliptic	$[-100,100]^D$	U	N	Y
f_{cec4}	Shift.Schwefel 1.2 Noise	$[-100,100]^D$	U	N	N
f_{cec5}	Schwefel 2.6 Opt on Bound	$[-100,100]^D$	U	N	N
f_{cec6}	Shift.Rosenbrock	$[-100,100]^D$	M	N	N
f_{cec7}	Shift.Ro.Griewank No Bound	$[0,600]^{D\dagger}$	M	N	Y
f_{cec8}	Shift.Ro.Ackley Opt on Bound	$[-32,32]^D$	M	N	Y
f_{cec9}	Shift.Rastrigin	$[-5,5]^D$	M	Y	N
f_{cec10}	Shift.Ro.Rastrigin	$[-5,5]^D$	M	N	Y
f_{cec11}	Shift.Ro.Weierstrass	$[-0.5,0.5]^D$	M	N	Y
f_{cec12}	Schwefel 2.13	$[-\pi,\pi]^D$	M	N	N
f_{cec13}	Griewank plus Rosenbrock	$[-3,1]^D$	M	N	N
f_{cec14}	Shift.Ro.Exp.Scaffer	$[-100,100]^D$	M	N	Y
f_{cec15}	Hybrid Composition	$[-5,5]^D$	M	N	N
f_{cec16}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec17}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec18}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec19}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec20}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec21}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec22}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec23}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec24}	Ro. Hybrid Composition	$[-5,5]^D$	M	N	Y
f_{cec25}	Ro. Hybrid Composition	$[2,5]^{D\dagger}$	M	N	Y
f_{soco1}	Shift.Sphere	$[-100,100]^D$	U	Y	N
f_{soco2}	Shift.Schwefel 2.21	$[-100,100]^D$	U	N	N
f_{soco3}	Shift.Rosenbrock	$[-100,100]^D$	M	N	N
f_{soco4}	Shift.Rastrigin	$[-5,5]^D$	M	Y	N
f_{soco5}	Shift.Griewank	$[-600,600]^D$	M	N	N
f_{soco6}	Shift.Ackley	$[-32,32]^D$	M	Y	N
f_{soco7}	Shift.Schwefel 2.22	$[-10,10]^D$	U	Y	N
f_{soco8}	Shift.Schwefel 1.2	$[-65.536,65.536]^D$	U	N	N
f_{soco9}	Shift.Extended f_{10}	$[-100,100]^D$	U	N	N
f_{soco10}	Shift.Bohachevsky	$[-15,15]^D$	U	N	N
f_{soco11}	Shift.Schaffer	$[-100,100]^D$	U	N	N
f_{soco12}	$f_{soco9} \oplus_{0.25} f_{soco1}$	$[-100,100]^D$	M	N	N
f_{soco13}	$f_{soco9} \oplus_{0.25} f_{soco3}$	$[-100,100]^D$	M	N	N
f_{soco14}	$f_{soco9} \oplus_{0.25} f_{soco4}$	$[-5,5]^D$	M	N	N
f_{soco15}	$f_{soco10} \oplus_{0.25} f_{soco7}$	$[-10,10]^D$	M	N	N
f_{soco16}	$f_{soco9} \oplus_{0.5} f_{soco1}$	$[-100,100]^D$	M	N	N
f_{soco17}	$f_{soco9} \oplus_{0.75} f_{soco3}$	$[-100,100]^D$	M	N	N
f_{soco18}	$f_{soco9} \oplus_{0.75} f_{soco4}$	$[-5,5]^D$	M	N	N
f_{soco19}	$f_{soco10} \oplus_{0.75} f_{soco7}$	$[-10,10]^D$	M	N	N

† denotes initialization range without bound constraints. Its global optimum is outside of initialization range.

Table 3. Results of the comparison between IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco over 25 independent runs for CEC'05 functions. Symbols $<$, $\approx$, and $>$ denote whether the performance of IPOP-CMA-ES-dp is statistically better, indifferent, or worse than that of IPOP-CMA-ES-tsoco according to the two-sided Wilcoxon matched-pairs signed-rank test at the 0.05 α-level. The numbers in parenthesis represent the times of $<$, $\approx$, and $>$, respectively. The numbers in parenthesis for ($<$, $=$, $>$) represent the times we have $<$, $=$, and $>$, respectively, when IPOP-CMA-ES-dp is compared with IPOP-CMA-ES-tsoco based on the mean or median errors.

f_{cec}	10 dimensions					30 dimensions					50 dimensions				
	IPOP-CMA-ES-dp Mean and Median			IPOP-CMA-ES-tsoco Mean and Median		IPOP-CMA-ES-dp Mean and Median			IPOP-CMA-ES-tsoco Mean and Median		IPOP-CMA-ES-dp Mean and Median			IPOP-CMA-ES-tsoco Mean and Median	
f_1	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_2	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_3	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_4	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	6.58E+02	1.75E+00	>	1.00E−08	1.00E−08	1.43E+04	1.27E+04	>	1.00E−08	1.00E−08
f_5	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	7.41E−02	8.61E−08	>	1.00E−08	1.00E−08
f_6	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_7	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_8	2.00E+01	2.00E+01	≈	2.02E+01	2.00E+01	2.04E+01	2.00E+01	<	2.08E+01	2.10E+01	2.09E+01	2.11E+01	≈	2.10E+01	2.11E+01
f_9	1.59E−01	1.00E−08	≈	4.81E−02	1.00E−08	1.87E+00	1.99E+00	≈	1.99E+00	1.99E+00	4.36E+00	3.98E+00	≈	4.18E+00	3.98E+00
f_{10}	3.18E−01	1.00E−08	≈	3.73E−03	1.00E−08	1.44E+00	9.95E−01	≈	1.59E+00	9.95E−01	2.89E+00	1.99E+00	≈	2.71E+00	2.98E+00
f_{11}	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	7.17E−02	1.00E−08	>	5.09E−05	1.00E−08	9.94E−02	1.00E−08	≈	6.03E−02	1.00E−08
f_{12}	4.07E+03	1.00E−08	>	1.00E−08	1.00E−08	1.19E+04	4.84E+03	>	4.22E+02	5.96E+01	4.25E+04	2.78E+04	>	4.69E+03	3.39E+03
f_{13}	6.49E−01	6.37E−01	≈	7.14E−01	6.94E−01	2.63E+00	2.71E+00	≈	2.53E+00	2.57E+00	4.44E+00	4.37E+00	<	4.70E+00	4.62E+00
f_{14}	1.96E+00	1.98E+00	≈	2.03E+00	2.09E+00	1.26E+01	1.26E+01	>	1.10E+01	1.13E+01	2.28E+01	2.30E+01	>	2.09E+01	2.12E+01
f_{15}	2.15E+02	2.00E+02	<	3.32E+02	4.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02
f_{16}	9.04E+01	9.11E+01	>	8.86E+01	9.03E+01	1.48E+01	1.51E+01	>	1.11E+01	1.07E+01	1.10E+01	1.14E+01	>	5.34E+00	5.47E+00
f_{17}	1.17E+02	1.09E+02	>	9.34E+01	9.43E+01	2.52E+02	1.80E+02	>	2.08E+02	5.60E+01	1.91E+02	1.62E+02	>	6.36E+01	4.99E+01
f_{18}	3.16E+02	3.00E+02	<	3.60E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.13E+02	9.16E+02	>	9.13E+02	9.13E+02
f_{19}	3.20E+02	3.00E+02	≈	3.20E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.13E+02	9.15E+02	>	9.13E+02	9.13E+02
f_{20}	3.20E+02	3.00E+02	≈	3.40E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.15E+02	9.15E+02	>	9.13E+02	9.13E+02
f_{21}	5.00E+02	5.00E+02	≈	5.00E+02	5.00E+02	5.00E+02	5.00E+02	≈	5.00E+02	5.00E+02	6.64E+02	5.00E+02	≈	7.05E+02	5.00E+02
f_{22}	7.28E+02	7.28E+02	≈	7.28E+02	7.28E+02	8.10E+02	8.11E+02	≈	8.17E+02	8.24E+02	8.19E+02	8.18E+02	≈	8.19E+02	8.21E+02
f_{23}	5.86E+02	5.59E+02	≈	5.59E+02	5.59E+02	5.34E+02	5.34E+02	≈	5.34E+02	5.34E+02	6.97E+02	5.40E+02	≈	7.30E+02	5.40E+02
f_{24}	2.33E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02
f_{25}	4.34E+02	4.04E+02	>	4.03E+02	4.03E+02	2.10E+02	2.10E+02	>	2.09E+02	2.09E+02	2.14E+02	2.14E+02	>	2.13E+02	2.13E+02
Mean	f_1-f_{25} (<, =, >): (6, 11, 8)					f_1-f_{25}(<, =, >): (4, 13, 8)					f_1-f_{25}(<, =, >): (4, 10, 11)†				
Median	f_1-f_{25} (<, =, >): (3, 19, 3)					f_1-f_{25}(<, =, >): (2, 16, 7)					f_1-f_{25}(<, =, >): (3, 12, 10)†				
By Func	f_1-f_{25} (<, ≈, >): (2, 19, 4)					f_1-f_{25}(<, ≈, >): (1, 17, 7)					f_1-f_{25}(<, ≈, >): (1, 14, 10)				
	10, 30 and 50 dimensional f_1-f_{25} (<, >):(5%, 28%)														

† denotes there is a significant difference over the distribution of mean or median errors between IPOP-CMA-ES-dp with IPOP-CMA-ES-tsoco by a two-sided Wilcoxon matched-pairs signed-ranks test at the 0,05 α-level.

Table 4. Comparison between IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco over 100 independent runs for CEC'05 functions. Symbols <, ≈, and > denote whether the performance of IPOP-CMA-ES-dp is statistically better, indifferent, or worse than that of IPOP-CMA-ES-tsoco according to the two-sided Wilcoxon matched-pairs signed-rank test at the 0.05 α-level. The numbers in parenthesis represent the times of <, ≈, and >, respectively. The numbers in parenthesis for (<, =, >) represent the times we have <, =, and >, respectively, when IPOP-CMA-ES-dp is compared with IPOP-CMA-ES-tsoco based on the mean or median errors.

f_{cec}	10 dimensions					30 dimensions					50 dimensions				
	IPOP-CMA-ES-dp Mean and Median			IPOP-CMA-ES-tsoco Mean and Median		IPOP-CMA-ES-dp Mean and Median			IPOP-CMA-ES-tsoco Mean and Median		IPOP-CMA-ES-dp Mean and Median			IPOP-CMA-ES-tsoco Mean and Median	
f_1	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_2	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_3	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_4	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.28E+03	1.35E+01	>	1.00E−08	1.00E−08	1.35E+04	1.16E+04	>	1.00E−08	1.00E−08
f_5	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	5.73E−02	4.71E−08	>	1.00E−08	1.00E−08
f_6	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_7	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_8	2.00E+01	2.00E+01	≈	2.02E+01	2.00E+01	2.05E+01	2.09E+01	<	2.07E+01	2.09E+01	2.08E+01	2.11E+01	≈	2.09E+01	2.11E+01
f_9	1.19E−01	1.00E−08	≈	7.70E−02	1.00E−08	1.92E+00	1.99E+00	≈	2.06E+00	1.99E+00	4.33E+00	3.98E+00	≈	4.19E+00	3.98E+00
f_{10}	1.29E−01	1.00E−08	≈	3.74E−02	1.00E−08	1.54E+00	1.01E+00	>	1.35E+00	9.95E−01	2.46E+00	1.99E+00	≈	2.31E+00	1.99E+00
f_{11}	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	2.51E−02	1.00E−08	>	4.21E−05	1.00E−08	9.15E−02	1.00E−08	≈	1.62E−04	1.00E−08
f_{12}	3.77E+03	1.34E−07	>	1.56E−02	1.00E−08	1.21E+04	5.81E+03	>	4.11E+02	6.73E+01	4.97E+04	3.87E+04	>	4.04E+03	2.95E+03
f_{13}	6.58E−01	6.73E−01	<	7.48E−01	7.14E−01	2.46E+00	2.54E+00	≈	2.55E+00	2.62E+00	4.47E+00	4.48E+00	<	4.60E+00	4.61E+00
f_{14}	2.00E+00	2.00E+00	≈	2.05E+00	2.06E+00	1.26E+01	1.27E+01	>	1.09E+01	1.12E+01	2.28E+01	2.30E+01	>	2.09E+01	2.12E+01
f_{15}	2.37E+02	2.00E+02	<	3.27E+02	4.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02
f_{16}	8.98E+01	9.10E+01	≈	8.96E+01	9.11E+01	1.48E+01	1.48E+01	>	1.22E+01	1.22E+01	1.09E+01	1.08E+01	>	5.97E+00	5.90E+00
f_{17}	1.15E+02	1.10E+02	>	9.48E+01	9.46E+01	2.36E+02	1.76E+02	>	1.52E+02	5.02E+01	1.93E+02	1.38E+02	>	7.10E+01	4.99E+01
f_{18}	3.10E+02	3.00E+02	<	3.45E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.15E+02	9.16E+02	>	9.13E+02	9.13E+02
f_{19}	3.20E+02	3.00E+02	≈	3.35E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.15E+02	9.15E+02	>	9.13E+02	9.13E+02
f_{20}	3.10E+02	3.00E+02	≈	3.25E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.15E+02	9.15E+02	>	9.13E+02	9.13E+02
f_{21}	5.00E+02	5.00E+02	≈	5.00E+02	5.00E+02	5.00E+02	5.00E+02	≈	5.00E+02	5.00E+02	6.46E+02	5.00E+02	≈	7.45E+02	5.00E+02
f_{22}	7.30E+02	7.28E+02	>	7.27E+02	7.27E+02	8.12E+02	8.12E+02	≈	8.15E+02	8.16E+02	8.20E+02	8.19E+02	≈	8.20E+02	8.21E+02
f_{23}	5.85E+02	5.59E+02	≈	5.59E+02	5.59E+02	5.34E+02	5.34E+02	≈	5.34E+02	5.34E+02	7.12E+02	5.43E+02	≈	7.64E+02	5.41E+02
f_{24}	2.08E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02
f_{25}	4.12E+02	4.05E+02	>	4.03E+02	4.04E+02	2.10E+02	2.10E+02	>	2.09E+02	2.09E+02	2.14E+02	2.14E+02	>	2.13E+02	2.13E+02
Mean	f_1-f_{25} (<, =, >): (7, 9, 9)					f_1-f_{25}(<, =, >): (4, 13, 8)					f_1-f_{25}(<, =, >): (4, 8, 13)†				
Median	f_1-f_{25} (<, =, >): (4, 17, 4)					f_1-f_{25}(<, =, >): (2, 16, 7)†					f_1-f_{25}(<, =, >): (2, 12, 11)†				
By Func	f_1-f_{25} (<, ≈, >): (3, 18, 4)					f_1-f_{25}(<, ≈, >): (1, 16, 8)					f_1-f_{25}(<, ≈, >): (1, 14, 10)				
	10, 30 and 50 dimensional f_1-f_{25} (<, >):(7%, 29%)														

† denotes there is a significant difference over the distribution of mean or median errors between IPOP-CMA-ES-dp with IPOP-CMA-ES-tsoco by a two-sided Wilcoxon matched-pairs signed-ranks test at the 0,05 α-level.

Table 5. Comparison between IPOP-CMA-ES-tcec and IPOP-CMA-ES-tsoco over 25 independent runs for CEC'05 functions. Symbols $<$, $\approx$, and $>$ denote whether the performance of IPOP-CMA-ES-tcec is statistically better, indifferent, or worse than that of IPOP-CMA-ES-tsoco according to the two-sided Wilcoxon matched-pairs signed-rank test at the 0.05 α-level. The numbers in parenthesis represent the times of $<$, $\approx$, and $>$, respectively. The numbers in parenthesis for $(<, =, >)$ represent the times we have $<$, $=$, and $>$, respectively, when IPOP-CMA-ES-dp is compared with IPOP-CMA-ES-tsoco based on the mean or median errors.

f_{cec}	10 dimensions					30 dimensions					50 dimensions				
	IPOP-CMA-ES-tcec Mean and Median			IPOP-CMA-ES-tsoco Mean and Median		IPOP-CMA-ES-tcec Mean and Median			IPOP-CMA-ES-tsoco Mean and Median		IPOP-CMA-ES-tcec Mean and Median			IPOP-CMA-ES-tsoco Mean and Median	
f_1	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_2	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_3	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_4	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	2.85E+02	1.00E−08	>	1.00E−08	1.00E−08
f_5	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	2.69E−08	2.72E−08	>	1.00E−08	1.00E−08	5.02E−08	4.85E−08	>	1.00E−08	1.00E−08
f_6	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	2.79E−02	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_7	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08
f_8	2.01E+01	2.00E+01	≈	2.02E+01	2.00E+01	2.08E+01	2.09E+01	≈	2.08E+01	2.10E+01	2.08E+01	2.11E+01	<	2.10E+01	2.11E+01
f_9	2.64E−01	1.00E−08	>	4.81E−02	1.00E−08	2.15E+00	1.99E+00	≈	1.99E+00	1.99E+00	5.49E+00	5.97E+00	>	4.18E+00	3.98E+00
f_{10}	1.59E−01	1.00E−08	>	3.73E−03	1.00E−08	1.67E+00	1.99E+00	>	1.59E+00	9.95E−01	3.54E+00	3.98E+00	>	2.71E+00	2.98E+00
f_{11}	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	8.69E−02	1.00E−08	>	5.09E−05	1.00E−08	1.83E−01	1.00E−08	≈	6.03E−02	1.00E−08
f_{12}	1.00E−08	1.00E−08	≈	1.00E−08	1.00E−08	4.88E+02	1.92E+02	≈	4.22E+02	5.96E+01	6.41E+03	4.59E+03	≈	4.69E+03	3.39E+03
f_{13}	6.14E−01	6.46E−01	<	7.14E−01	6.94E−01	2.48E+00	2.50E+00	≈	2.53E+00	2.57E+00	4.42E+00	4.47E+00	<	4.70E+00	4.62E+00
f_{14}	8.07E−01	6.66E−01	<	2.03E+00	2.09E+00	9.93E+00	9.94E+00	<	1.10E+01	1.13E+01	2.02E+01	1.98E+01	<	2.09E+01	2.12E+01
f_{15}	2.96E+02	3.00E+02	<	3.32E+02	4.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02
f_{16}	8.83E+01	8.94E+01	≈	8.86E+01	9.03E+01	1.06E+01	1.04E+01	≈	1.11E+01	1.07E+01	9.46E+00	9.09E+00	>	5.34E+00	5.47E+00
f_{17}	1.20E+02	1.17E+02	>	9.34E+01	9.43E+01	2.14E+02	1.52E+02	≈	2.08E+02	5.60E+01	9.55E+01	8.06E+01	>	6.36E+01	4.99E+01
f_{18}	3.00E+02	3.00E+02	<	3.60E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.15E+02	9.16E+02	>	9.13E+02	9.13E+02
f_{19}	3.00E+02	3.00E+02	≈	3.20E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.16E+02	9.16E+02	>	9.13E+02	9.13E+02
f_{20}	3.00E+02	3.00E+02	≈	3.40E+02	3.00E+02	9.04E+02	9.04E+02	≈	9.04E+02	9.04E+02	9.15E+02	9.14E+02	>	9.13E+02	9.13E+02
f_{21}	5.00E+02	5.00E+02	≈	5.00E+02	5.00E+02	5.00E+02	5.00E+02	≈	5.00E+02	5.00E+02	9.68E+02	1.01E+03	>	7.05E+02	5.00E+02
f_{22}	7.27E+02	7.26E+02	≈	7.28E+02	7.28E+02	8.10E+02	8.07E+02	≈	8.17E+02	8.24E+02	8.15E+02	8.14E+02	≈	8.19E+02	8.21E+02
f_{23}	5.59E+02	5.59E+02	≈	5.59E+02	5.59E+02	5.34E+02	5.34E+02	≈	5.34E+02	5.34E+02	9.04E+02	1.02E+03	>	7.30E+02	5.40E+02
f_{24}	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02	2.00E+02	2.00E+02	≈	2.00E+02	2.00E+02
f_{25}	4.04E+02	4.04E+02	>	4.03E+02	4.03E+02	2.09E+02	2.09E+02	>	2.09E+02	2.09E+02	2.13E+02	2.13E+02	>	2.13E+02	2.13E+02
Mean	f_1-f_{25} $(<, =, >)$: (9, 12, 4)					f_1-$f_{25}$$(<, =, >)$: (4, 14, 7)					f_1-$f_{25}$$(<, =, >)$: (4, 8, 14)†				
Median	f_1-f_{25} $(<, =, >)$: (5, 18, 2)					f_1-$f_{25}$$(<, =, >)$: (5, 16, 4)					f_1-$f_{25}$$(<, =, >)$: (3, 11, 11)†				
By Func	f_1-f_{25} $(<, \approx, >)$: (4, 17, 4)					f_1-$f_{25}$$(<, \approx, >)$: (1, 20, 4)					f_1-$f_{25}$$(<, \approx, >)$: (3, 10, 12)				
	10, 30 and 50 dimensional f_1-f_{25} $(<, >)$:(11%, 27%)														

† denotes there is a significant difference over the distribution of mean or median errors between IPOP-CMA-ES-tcec with IPOP-CMA-ES-tsoco by a two-sided Wilcoxon matched-pairs signed-ranks test at the 0,05 α-level.

Table 6. The average errors obtained by MA-LSch-CMA, PS-CMA-ES and IPOP-CMA-ES-tsoco (MA, PS, IPOP-tsoco for their abbreviations, respectively, in this table) over 25 independent runs for CEC'05 functions. The numbers in parenthesis represent the times of <, =, and >, respectively, when the corresponding algorithms are compared with IPOP-CMA-ES-tsoco based on the average errors. The number of optima found by each algorithm and the average ranking of each algorithm are also given.

f_{cec}	10 dimensions			30 dimensions			50 dimensions		
	Mean Errors			Mean Errors			Mean Errors		
	MA	PS	IPOP-tsoco	MA	PS	IPOP-tsoco	MA	PS	IPOP-tsoco
f_1	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08
f_2	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08	1.00E−08	3.06E−02	7.36E−06	1.00E−08
f_3	1.00E−08	1.45E−01	1.00E−08	2.75E+04	2.96E+04	1.00E−08	3.21E+04	9.10E+04	1.00E−08
f_4	5.54E−03	1.00E−08	1.00E−08	3.02E+02	4.56E+03	1.00E−08	3.23E+03	2.17E+04	1.00E−08
f_5	6.75E−07	1.00E−08	1.00E−08	1.26E+03	2.52E+01	1.00E−08	2.69E+03	1.79E+03	1.00E−08
f_6	3.19E−01	1.00E−08	1.00E−08	1.12E+00	1.15E+01	1.00E−08	4.10E+00	2.91E+01	1.00E−08
f_7	1.43E−01	1.00E−08	1.00E−08	1.75E−02	1.00E−08	1.00E−08	5.40E−03	1.00E−08	1.00E−08
f_8	2.00E+01	2.00E+01	2.02E+01	2.00E+01	2.00E+01	2.08E+01	2.00E+01	2.00E+01	2.10E+01
f_9	1.00E−08	3.98E−02	4.81E−02	1.00E−08	8.76E−01	1.99E+00	1.00E−08	5.45E+00	4.18E+00
f_{10}	2.67E+00	1.00E−08	3.73E−03	2.25E+01	5.57E−01	1.59E+00	5.01E+01	5.33E+00	2.71E+00
f_{11}	2.43E+00	8.51E−01	1.00E−08	2.15E+01	7.10E+00	5.09E−05	4.13E+01	1.59E+01	6.03E−02
f_{12}	1.14E+02	1.10E+00	1.00E−08	1.67E+03	8.80E+02	4.22E+02	1.39E+04	6.90E+03	4.69E+03
f_{13}	5.45E−01	3.67E−01	7.14E−01	2.03E+00	2.05E+00	2.53E+00	3.15E+00	4.15E+00	4.70E+00
f_{14}	2.25E+00	3.40E+00	2.03E+00	1.25E+01	1.24E+01	1.10E+01	2.22E+01	2.15E+01	2.09E+01
f_{15}	2.24E+02	8.67E+01	3.32E+02	3.00E+02	1.37E+02	2.00E+02	3.72E+02	1.25E+02	2.00E+02
f_{16}	9.18E+01	9.28E+01	8.86E+01	1.26E+02	1.59E+01	1.11E+01	6.90E+01	1.62E+01	5.34E+00
f_{17}	1.01E+02	1.12E+02	9.34E+01	1.83E+02	9.15E+01	2.08E+02	1.47E+02	9.13E+01	6.36E+01
f_{18}	8.84E+02	3.60E+02	3.60E+02	8.98E+02	9.05E+02	9.04E+02	9.41E+02	8.70E+02	9.13E+02
f_{19}	8.78E+02	3.25E+02	3.20E+02	9.01E+02	8.85E+02	9.04E+02	9.38E+02	9.13E+02	9.13E+02
f_{20}	8.63E+02	3.43E+02	3.40E+02	8.96E+02	9.05E+02	9.04E+02	9.28E+02	9.09E+02	9.13E+02
f_{21}	7.94E+02	4.71E+02	5.00E+02	5.12E+02	5.00E+02	5.00E+02	5.00E+02	6.62E+02	7.05E+02
f_{22}	7.53E+02	7.46E+02	7.28E+02	8.80E+02	8.43E+02	8.17E+02	9.14E+02	8.63E+02	8.19E+02
f_{23}	8.88E+02	5.58E+02	5.59E+02	5.34E+02	5.34E+02	5.34E+02	5.39E+02	8.12E+02	7.30E+02
f_{24}	2.28E+02	2.00E+02	2.00E+02	2.00E+02	2.00E+02	2.00E+02	2.00E+02	2.00E+02	2.00E+02
f_{25}	4.55E+02	4.00E+02	4.03E+02	2.14E+02	2.10E+02	2.09E+02	2.21E+02	2.14E+02	2.13E+02
V.S.	(3, 4, 18)†	(8, 8, 9)		(7, 4, 14)†	(7, 6, 12)		(5, 2, 18)†	(6, 4, 15)†	
Optima	4	7	**9**	3	3	**7**	2	2	**7**
Rank	2.52	1.78	**1.7**	2.26	1.98	**1.76**	2.42	2.02	**1.56**

† denotes there is a significant difference over the distribution of average errors between the corresponding algorithm with IPOP-CMA-ES-tsoco by a two-sided Wilcoxon matched-pairs signed-ranks test at the 0,05 α-level.

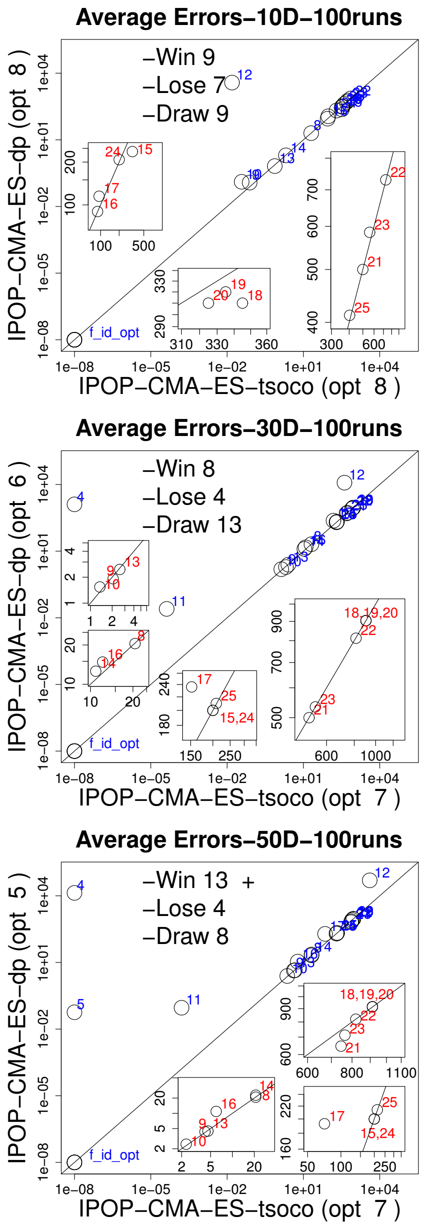

Fig. 2. Correlation plots of IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco on dimensions 10, 30 and 50 respectively. Each point represents the average error value obtained by either of the two algorithms. A point on the upper triangle delimited by the diagonal indicates better performance for the algorithm on the x-axis; a point on the lower right triangle indicates better performance for the algorithm on the y-axis. The number labeled beside some outstanding points represent the index of the corresponding function. The comparison is conducted based on average error values and the comparison results of the algorithm on the x-axis are presented in form of -win, -draw, -lose, respectively. We marked with a + symbol those cases in which there is a statistically significant difference at the 0.05 α-level between the algorithms. The number of opt on the axes shows the number of averages lower than the zero threshold by the corresponding algorithm.

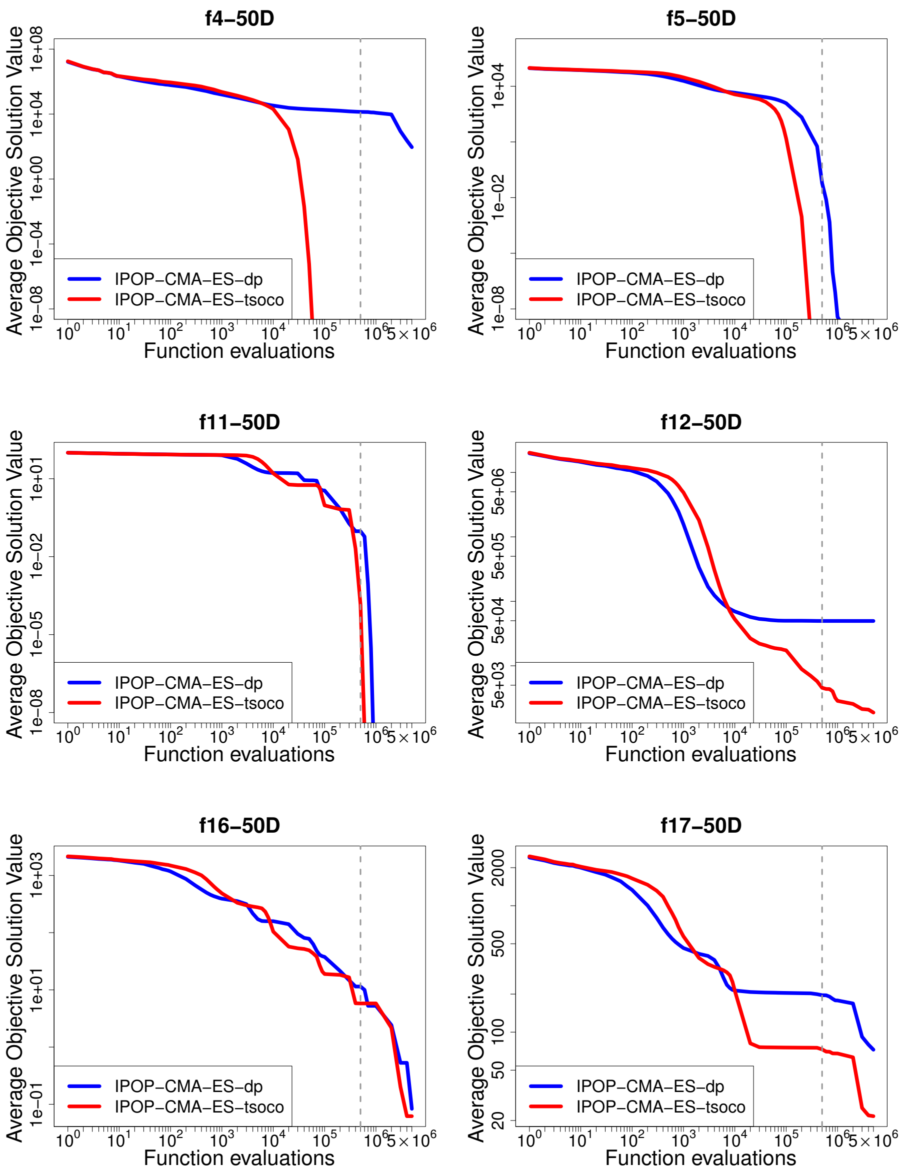

Fig. 3. The development of the average error of the fitness values across 100 independent runs of IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco over the number of function evaluations on functions f_{cec4}, f_{cec5}, f_{cec11}, f_{cec12}, f_{cec16} and f_{cec17} of 50 dimensions. The vertical, dotted line in each plot indicates 5.00E+05 function evaluations, which is the termination criterion for the number of function evaluations in the CEC'05 protocol.

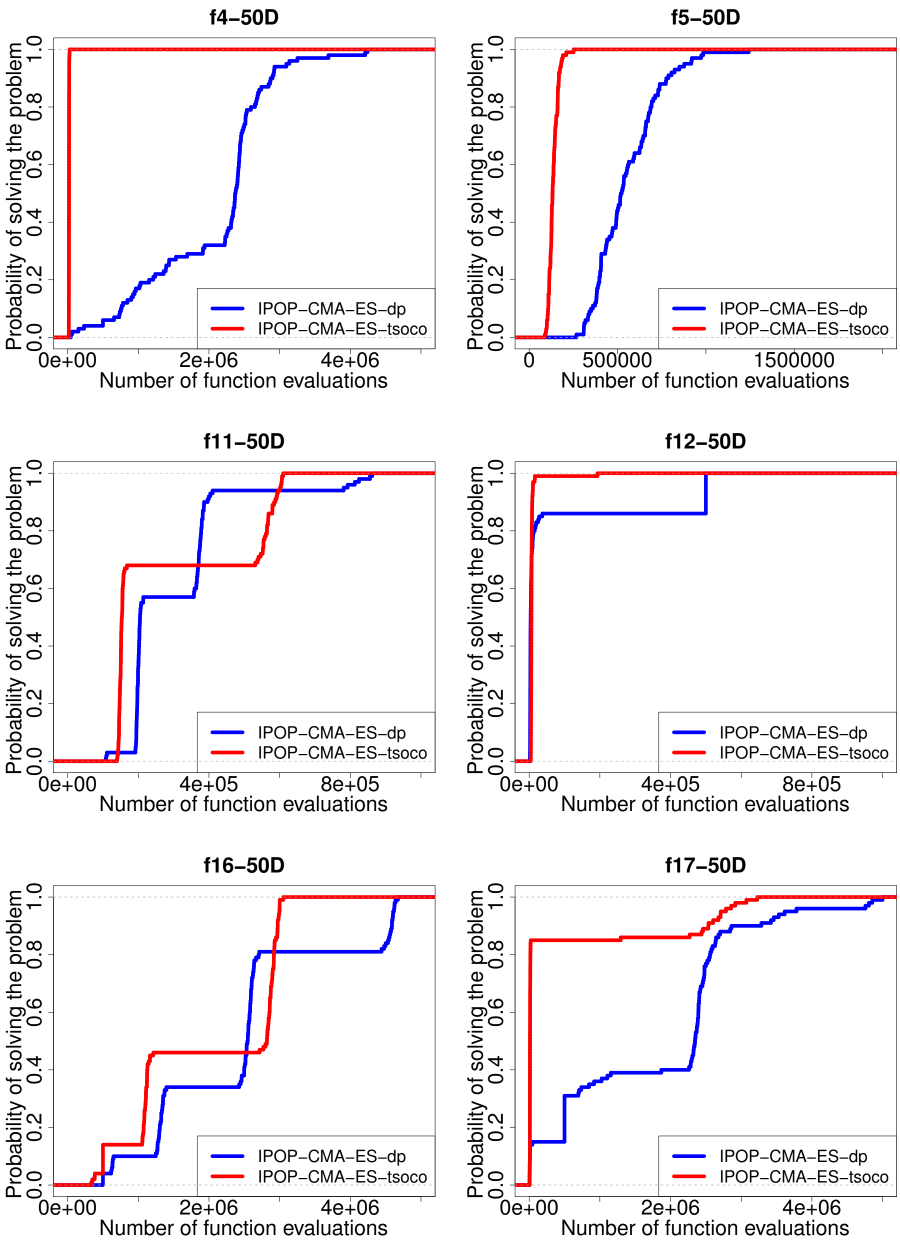

Fig. 4. The qualified run-length distributions (RLDs, for short) over 100 independent runs obtained by IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco on the 50 dimensional versions of functions f_{cec4}, f_{cec5}, f_{cec11}, f_{cec12}, f_{cec16} and f_{cec17}. The required solution qualities are 1.00E+03, 1.00E−08, 1.00E−08, 1.00E+05, 1.00E+00 and 1.00E+02, respectively.

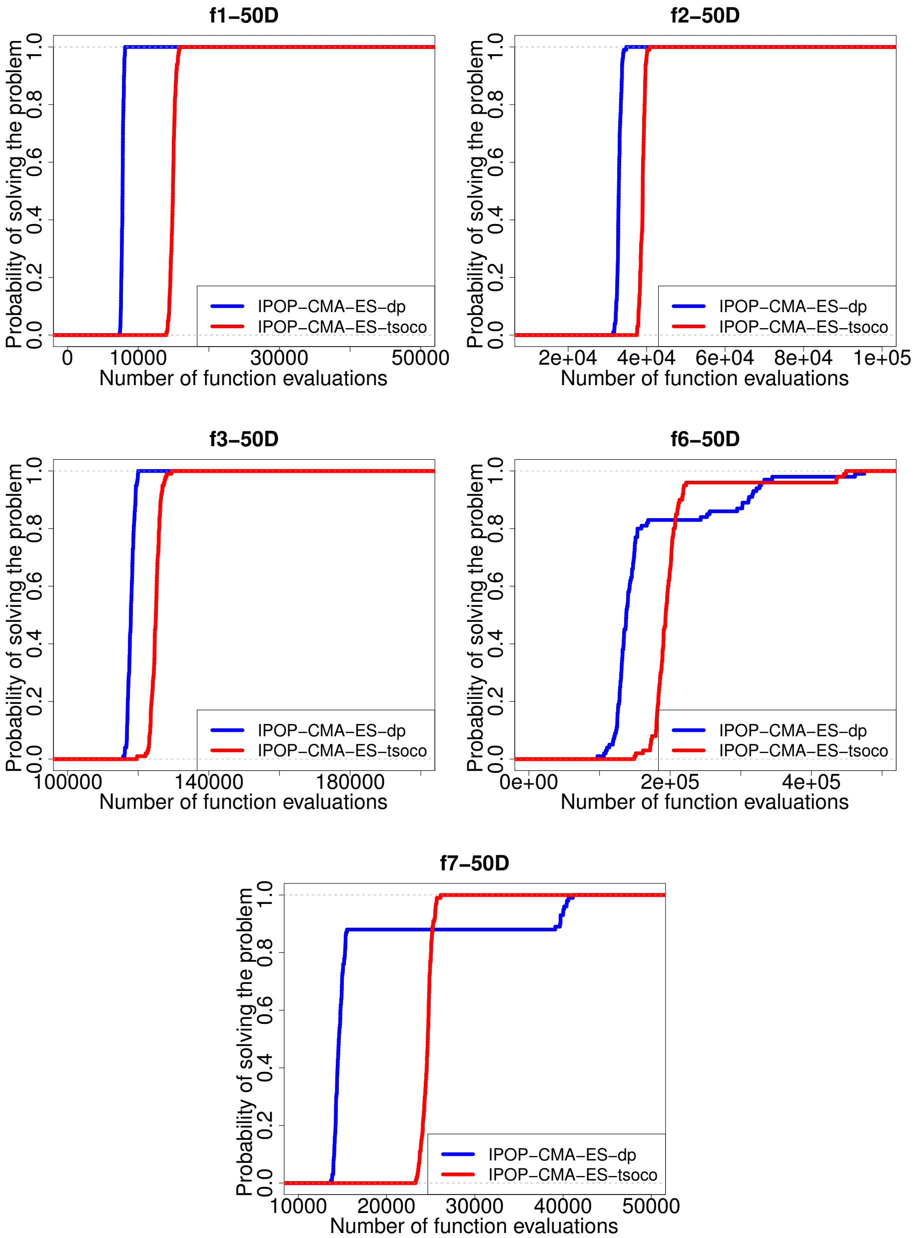

Fig. 5. The qualified run-length distributions (RLDs, for short) over 100 independent runs obtained by IPOP-CMA-ES-dp and IPOP-CMA-ES-tsoco on the 50 dimensional versions of functions f_{cec1}, f_{cec2}, f_{cec3}, f_{cec6} and f_{cec7}. The solution quality demanded is 1.00E−08 for each function.

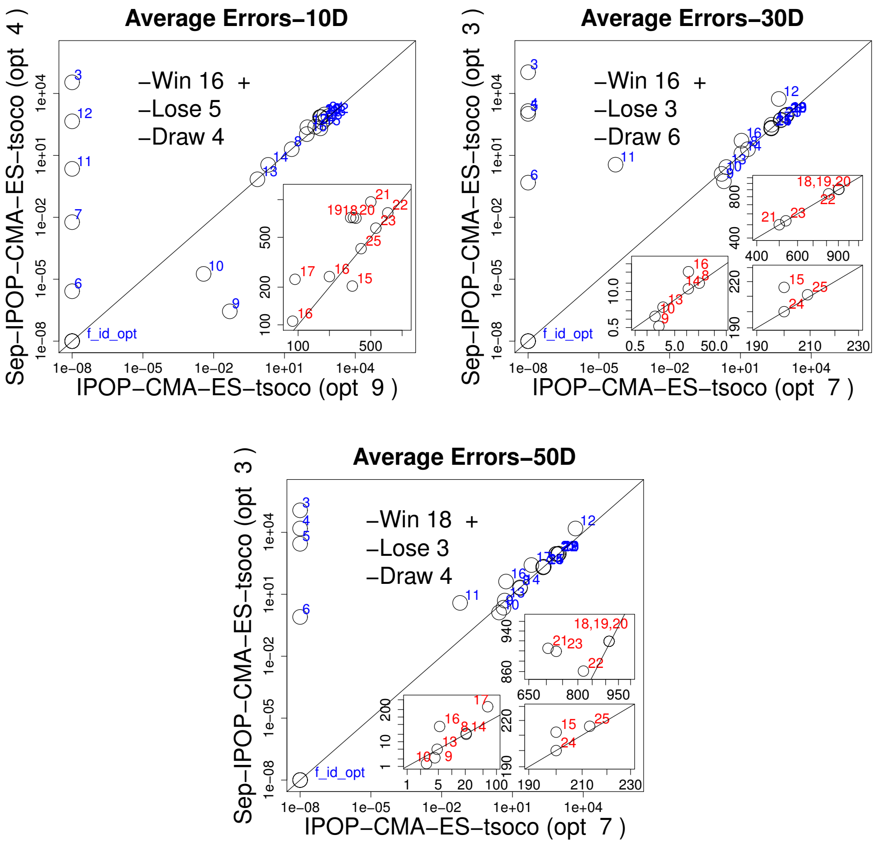

Fig. 6. Correlation plots of IPOP-CMA-ES-tsoco and Sep-IPOP-CMA-ES-tsoco on dimensions 10, 30 and 50 respectively. Each point represents the average error value over 25 independent runs obtained by either of the two algorithms. A point on the upper triangle delimited by the diagonal indicates better performance for the algorithm on the x-axis; a point on the lower right triangle indicates better performance for the algorithm on the y-axis. The number labeled beside some outstanding points represent the index of the corresponding function. The comparison is conducted based on average error values and the comparison results of the algorithm on the x-axis are presented in the form of -win, -draw, -lose, respectively, using IPOP-CMA-ES-tsoco as the reference. We marked with a + symbol those cases in which there is a statistically significant difference at the 0.05 α-level between the algorithms. The number of opt on the axes shows the number of optima, that is averages lower than the zero threshold, obtained by the corresponding algorithm.

Table 7. Summary of the comparison with IPOP-CMA-ES-tsoco on 10, 30 and 50 dimensions w.r.t. average error values: (better, equal, worse). Error values lower than 10^{-8} are approximated to 10^{-8}.

VS. IPOP-CMA-ES-tsoco	IPOP-CMA-ES-05	Sep-IPOP-CMA-ES-tsoco
10 Dim	(6, 10, 9)	$(5, 4, 16)^{\dagger}$
30 Dim	$(4, 11, 10)^{\dagger}$	$(3, 6, 16)^{\dagger}$
50 Dim	(7, 6, 12)	$(3, 4, 18)^{\dagger}$

† denotes there is a significant difference over the distribution of average errors between the corresponding algorithm and IPOP-CMA-ES-tsoco according to a two-sided Wilcoxon matched-pairs signed-rank test at the 0,05 α-level.