

Benchmarking sep-CMA-ES on the BBOB-2009 Noisy Testbed

Raymond Ros
Univ. Paris-Sud, LRI
UMR 8623 / INRIA Saclay, projet TAO
F-91405 Orsay, France
raymond.ros@lri.fr

ABSTRACT

A partly time and space linear CMA-ES is benchmarked on the BBOB-2009 noisy function testbed. This algorithm with a multistart strategy with increasing population size solves 10 functions out of 30 in 20-D.

Categories and Subject Descriptors

G.1.6 [Numerical Analysis]: Optimization—*global optimization, unconstrained optimization*; F.2.1 [Analysis of Algorithms and Problem Complexity]: Numerical Algorithms and Problems

General Terms

Algorithms

Keywords

Benchmarking, Black-box optimization, Evolutionary computation, Covariance matrix adaptation, Evolution strategy

1. INTRODUCTION

The sep-CMA-ES algorithm introduced in [7] is a variant of the covariance matrix adaptation evolution strategy (CMA-ES) [5] that is linear in time and space. This property combined with a faster learning rate makes sep-CMA-ES appropriate for separable function and larger dimensions. A mixed strategy of using sep-CMA-ES and CMA-ES is proposed here and benchmarked on a noisy function testbed.

2. ALGORITHM PRESENTATION

In its design, the sep-CMA-ES differs from the CMA-ES by two aspects: first, the covariance matrix is constrained to be diagonal at each of its update, second, the learning rate is increased by a factor of $\frac{n+3/2}{3}$, where n is the dimension of the search space¹. These modifications result

¹Please note that the factor for the learning rate is smaller than the one in [7].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

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

in an algorithm that trades model complexity with a time and space scaling that is better than the original CMA-ES. The $(\mu/\mu_w, \lambda)$ -sep-CMA-ES has been shown to outperform $(\mu/\mu_w, \lambda)$ -CMA-ES on separable functions.

We propose here what would be the best of two worlds: to use sep-CMA-ES for the first few iterations and then switch to CMA-ES. At the time of the switch, all parameters are retained except for the learning rate that is decreased back to its default value. This implies the diagonal covariance matrix acquired using sep-CMA-ES is directly used by CMA-ES. This mixed strategy is therefore expected to be faster than CMA-ES on separable functions. Ongoing work has also shown that for some test functions the first iterations using sep-CMA-ES would not disadvantage the latter use of CMA-ES in any way. In other terms, the cost of initially using sep-CMA-ES would not induce a penalty in the cost of solving the function with CMA-ES afterwards. The author admits some functions could induce such a penalty.

As for the multistart strategy, we use the increasing population size IPOPOP-CMA-ES [1]. Though this approach has shown its limits [6], independent restart may improve the probability of the algorithm reaching a given target function value.

3. EXPERIMENTAL PROCEDURE

The Matlab implementation of the CMA-ES (version 3.23beta) is used². We use the $(\mu/\mu_w, \lambda)$ -IPOPOP-CMA-ES variant with an initial default population size $\lambda = 4 + \lfloor 3 \ln(n) \rfloor$ increasing twice at each restart. Except the learning rate, all other algorithm parameters are set to their default values. The covariance matrix is constrained to be diagonal only for the first $1 + 100n/\sqrt{\lambda}$ iterations of the *first start*. A maximum of 8 independent restarts is conducted. Restarts occur after $100 + 300n\sqrt{n/\lambda}$ iterations or if any of the default stopping criterion is met. The initial stepsize has been set to 2 and the starting point has been chosen uniformly in $[-4, 4]^n$. The maximum number of function evaluations was set to 10^4 times the dimension. No parameter tuning was done, the CrE [3] is computed to zero.

4. RESULTS

Results from experiments according to [3] on the benchmarks functions given in [2, 4] are presented in Figures 1 and 2 and in Tables 1 and 2. From the results of this algorithm, the uniform noise model is the most difficult to deal

²Latest version available here:<http://www.lri.fr/~hansen/cmaesintro.html>

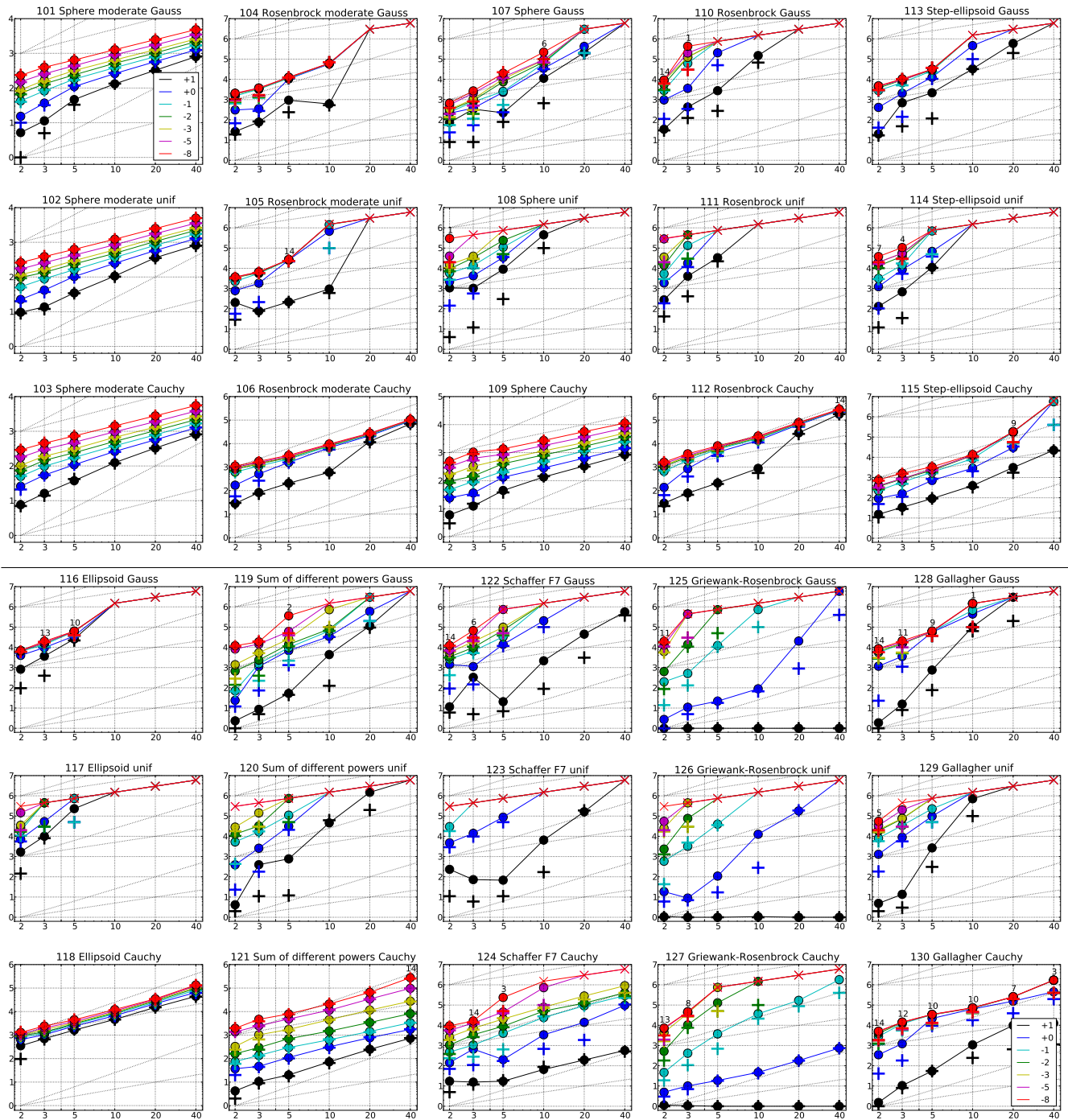


Figure 1: Expected Running Time (ERT, ●) to reach $f_{\text{opt}} + \Delta f$ and median number of function evaluations of successful trials (+), shown for $\Delta f = 10, 1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-5}, 10^{-8}$ (the exponent is given in the legend of f_{101} and f_{130}) versus dimension in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

f_{101} in 5-D, N=15, mFE=738					f_{101} in 20-D, N=15, mFE=2654					f_{102} in 5-D, N=15, mFE=754					f_{102} in 20-D, N=15, mFE=2798						
Δf	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	#	ERT	10%	90%	RT _{succ}	
10	15	4.6e1	3.7e1	5.6e1	4.6e1	15	3.2e2	3.1e2	3.4e2	3.2e2	15	3.5e1	2.9e1	4.1e1	3.5e1	15	3.6e2	3.4e2	3.8e2	3.6e2	
1	15	1.1e2	1.0e2	1.2e2	1.1e2	15	5.8e2	5.6e2	6.0e2	5.8e2	1	15	1.0e2	9.3e1	1.1e2	1.0e2	15	5.8e2	5.6e2	6.0e2	5.8e2
1e-1	15	1.8e2	1.7e2	1.9e2	1.8e2	15	8.0e2	7.8e2	8.2e2	8.0e2	1e-1	15	1.6e2	1.5e2	1.7e2	1.6e2	15	8.2e2	7.9e2	8.5e2	8.2e2
1e-3	15	3.2e2	3.1e2	3.3e2	3.2e2	15	1.3e3	1.2e3	1.3e3	1.3e3	1e-3	15	3.0e2	2.9e2	3.1e2	3.0e2	15	1.3e3	1.3e3	1.3e3	1.3e3
1e-5	15	4.4e2	4.3e2	4.5e2	4.4e2	15	1.7e3	1.7e3	1.8e3	1.7e3	1e-5	15	4.3e2	4.2e2	4.4e2	4.3e2	15	1.8e3	1.7e3	1.8e3	1.8e3
1e-8	15	6.4e2	6.2e2	6.6e2	6.4e2	15	2.5e3	2.4e3	2.5e3	2.5e3	1e-8	15	6.3e2	6.1e2	6.5e2	6.3e2	15	2.5e3	2.5e3	2.5e3	2.5e3

Table 1: Shown are, for functions f_{101} - f_{120} and for a given target difference to the optimal function value Δf : the number of successful trials (#); the expected running time to surpass $f_{\text{opt}} + \Delta f$ (ERT, see Figure 1); the 10%-tile and 90%-tile of the bootstrap distribution of ERT; the average number of function evaluations in successful trials or, if none was successful, as last entry the median number of function evaluations to reach the best function value (RT_{succ}). If $f_{\text{opt}} + \Delta f$ was never reached, figures in *italics* denote the best achieved Δf -value of the median trial and the 10% and 90%-tile trial. Furthermore, N denotes the number of trials, and mFE denotes the maximum of number of function evaluations executed in one trial. See Figure 1 for the names of functions.

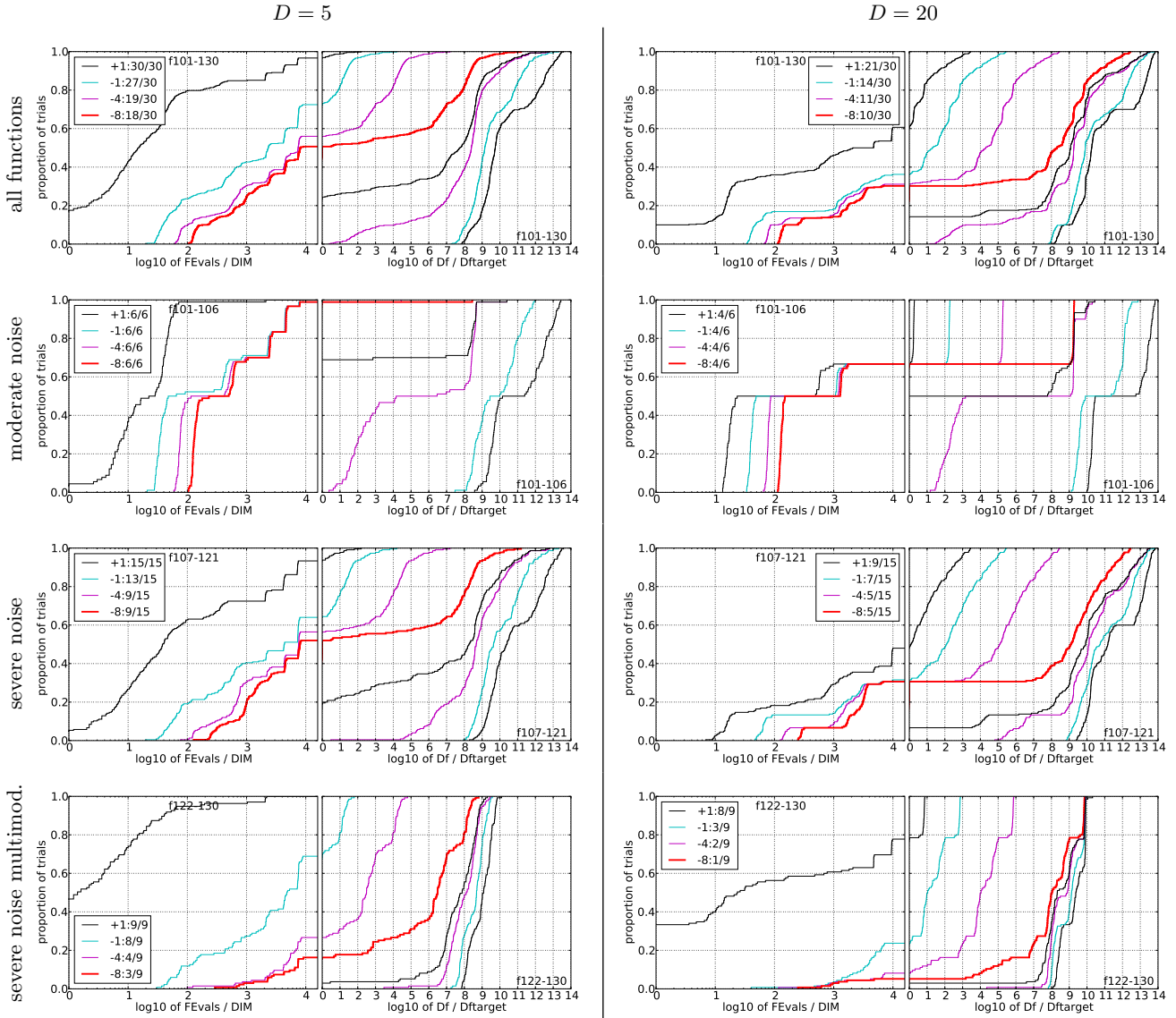


Figure 2: Empirical cumulative distribution functions (ECDFs), plotting the fraction of trials versus running time (left subplots) or versus Δf (right subplots). The thick red line represents the best achieved results. Left subplots: ECDF of the running time (number of function evaluations), divided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

