Benchmarking a BI-Population CMA-ES on the BBOB-2009 Noisy Testbed

Nikolaus Hansen

To cite this version:

HAL Id: inria-00382101
https://inria.hal.science/inria-00382101
Submitted on 7 May 2009

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.
Benchmarking a BI-Population CMA-ES on the BBOB-2009 Noisy Testbed

Nikolaus Hansen
Microsoft Research–INRIA Joint Centre
28 rue Jean Rostand
91893 Orsay Cedex, France
Nikolaus.Hansen@inria.fr

ABSTRACT
We benchmark the BI-population CMA-ES on the BBOB-2009 noisy functions testbed. BI-population refers to a multistart strategy with equal budgets for two interlaced restart strategies, one with an increasing population size and one with varying small population sizes. The latter is presumably of little use on a noisy testbed. The BI-population CMA-ES could solve 29, 27 and 26 out of 30 functions in search space dimension 5, 10 and 20 respectively. The time to find the solution ranges between 100D and 10^5D^2 objective function evaluations, where D is the search space dimension.

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, Direct search, Evolutionary computation, CMA-ES

1. INTRODUCTION
The covariance matrix adaptation evolution strategy (CMA-ES) [2, 6, 7] is a stochastic, population-based search method in continuous search spaces, aiming at minimizing an objective function \( f: \mathbb{R}^D \rightarrow \mathbb{R} \) in a black-box scenario. In this paper, the \((\mu/\mu_w, \lambda)\)-CMA-ES is applied in a multistart strategy and benchmarked on 30 noisy functions. The multistart consists of two interlacing strategies, one with increasing population size, the other with varying small population size. The algorithm is given in a complementing paper in the same proceedings [3].

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.

2. ALGORITHM AND PARAMETER SETTINGS
The algorithm and all parameters are described in [3]. The following parameters have been chosen differently, where \( \lambda \) denotes the offspring population size.

MaxIter = 1000 + 500(D + 3)^2/\sqrt{\lambda} has been chosen ten times larger, as the noisy functions are expected to be more difficult to solve.

\[
c_1 = \frac{0.4}{2^{t+1}c_{10}^+c_{30}^+} \quad \text{and} \quad c_\mu = \min \left( 1, \frac{0.4}{(D+1)^2/\sqrt{\mu w}} \right),
\]

the learning rates for the covariance matrix have been chosen five times smaller than by default. With default learning rates, the learning of the covariance matrix is the algorithm component that is most susceptible to uncertainties in the selection. Surprisingly, this change did not produce a striking improvement, but slightly more consistent results (e.g. \( f_{105}, f_{108} \) and \( f_{117} \) are solved also in 40-D).

Restarts are launched until \( 10^6D \) function evaluations are exceeded. The stagnation termination criterion

Stagnation: terminate a run, if the median of the 20 newest values is not smaller than the median of the 20 oldest values, respectively, in the two arrays containing the best function values and the median function values of the last \( 0.2 t + 120 + 30D/\lambda \) iterations, where \( t \) denotes the iteration counter,

turns out to be crucial for the noisy testbed. Most other standard termination criteria regularly fail. We presume that restarts with small population size are less valuable on a noisy testbed, which leaves yet room for improvement.

The same parameter setting is used for all functions and therefore the crafting effort according to [4] is CrE = 0.

3. SUCCESSFUL POPULATION SIZE
We investigate the population sizes of the final successful runs. In Table 1 minimal, median (the larger in case of even data) and maximal population size are given for 10-D and 20-D. Functions solved with default offspring population size, \( \lambda = 10 \) and 12, are the sphere function with moderate noise and about half of the functions with Cauchy noise (function numbers in italics). Otherwise is the typical population size \( 10D \) or larger.

Table 2 tabulates minimal, median (the larger in case of even data) and maximal initial step-size \( \sigma^0 \) of the final successful runs, whenever \( \sigma^0 < 2 \) in at least one successful case.
4. RESULTS AND DISCUSSION

The results of CPU timing experiments are given in [3]: using Matlab, about $2\times10^{-4}$ seconds per function evaluation are needed on the benchmarks functions given in [1, 5] are presented in Figures 1 and 2 and in Tables 3 and 4. The typical scaling of the running time (number of function evaluations) is quadratic with the dimension (see Figure 1). With the moderate noise models and with the Cauchy noise model the sphere function can be solved in linear time.

More severe noise impairs the scaling behavior by one order of magnitude to quadratic. Similarly, on the Rosenbrock function the severe noise impairs the scaling by one order from roughly quadratic to roughly cubic. Here, the observed failure is presumably due to a too small maximum number of function evaluations allowed.

The expected running times appear rather uniform on the log-scale between $10^0 D$ and $10^{5} D^2$ function evaluations (upper figures in Figure 2). The graphs suggest that, in higher dimension, with more function evaluations even more functions can be solved.

Acknowledgments

The author would like to acknowledge the great and hard work of the BBOB team with particular kudos to Raymond Ros, Steffen Finck and Anne Auger, and Anne Auger and Marc Schoenauer for their kind and persistent support.

5. REFERENCES


Figure 1: Expected Running Time (ERT; ●) to reach $f_{\text{opt}} + \Delta f$ and median number of function evaluations of successful trials (+), shown for $\Delta f = 10, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-5}, 10^{-8}$ (the exponent is given in the legend of $f_{101}$ and $f_{130}$) versus dimension in log-log presentation. The ERT($\Delta f$) equals $\#\text{FEs}(\Delta f)$ divided by the number of successful trials, where a trial is successful if $f_{\text{opt}} + \Delta f$ was surpassed during the trial. The $\#\text{FEs}(\Delta f)$ 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 $f_{\text{opt}}$ denotes the optimal function value. Crosses (×) indicate the total number of function evaluations $\#\text{FEs}(-\infty)$. 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.
<table>
<thead>
<tr>
<th>$\Delta t$</th>
<th>#</th>
<th>$%$ ERT</th>
<th>10%</th>
<th>90%</th>
<th>RT</th>
<th>$\Delta f$</th>
<th>#</th>
<th>$%$ ERT</th>
<th>10%</th>
<th>90%</th>
<th>RT</th>
<th>$\Delta f$</th>
<th>#</th>
<th>$%$ ERT</th>
<th>10%</th>
<th>90%</th>
<th>RT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
<td>2.0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
<td>3.0</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
<td>4.0</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
<td>6.0</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
<td>8.0</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
<td>9.0</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
<td>10.0</td>
</tr>
</tbody>
</table>

Table 3: Showed are, for functions $f_{101}-f_{20}$ and for a given target difference to the optimal function value $\Delta f$: the number of successful trials (\#); the expected running time to surpass $f_{opt} + \Delta f$ (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 isSuccess). If $f_{opt} + \Delta f$ was never reached, figures in italics denote the best achieved $\Delta$-value of the median trial. If $f_{opt} + \Delta f$ exceeds the 10% and 90%-tile, the function is not considered. N denotes the number of trials and mFE denotes the maximum 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 $\Delta f$ (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 $f_{opt} + \Delta f$ with $\Delta f = 10^k$, where $k$ is the first value in the legend. Right subplots: ECDF of the best achieved $\Delta f$ divided by $10^k$ (upper left lines in continuation of the left subplot), and best achieved $\Delta f$ divided by $10^{-8}$ for running times of $D, 10D, 100D, \ldots$ 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 $\text{DIM}$ denote search space dimension, and $\Delta f$ and $Df$ denote the difference to the optimal function value.
Table 4: Show are, for functions $f_{121}$-$f_{130}$ and for a given target difference to the optimal function value $\Delta f$: 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 expected running time 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$_{\text{succ}}$). If $f_{\text{opt}} + \Delta f$ was never reached, figures in italics denote the best achieved $\Delta f$-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.