Improving Evolution Strategies through Active Covariance Matrix Adaptation

Grahame A. Jastrebski and Dirk V. Arnold

Abstract— **This paper proposes a novel modification to the derandomised covariance matrix adaptation algorithm used in connection with evolution strategies. In existing variants of that algorithm, only information gathered from successful offspring candidate solutions contributes to the adaptation of the covariance matrix, while old information passively decays. We propose to use information about unsuccessful offspring candidate solutions in order to actively reduce variances of the mutation distribution in unpromising directions of the search space. The resulting strategy is referred to as Active-CMA-ES. In experiments on a standard suite of test functions, Active-CMA-ES consistently outperforms other strategy variants.**

I. Introduction

Evolution strategies are evolutionary algorithms that are most commonly used for the optimisation of real-valued functions $f : I\!R^n \rightarrow I\!R$. See [1] for a comprehensive introduction. Candidate solutions are n-dimensional vectors of real numbers. Variation and selection are iterated with the goal of evolving populations of higher and higher quality. The quality of candidate solutions is determined by the objective function, and selection chooses good candidate solutions as parents for future generations while inferior ones are discarded. Variation consists of recombination and mutation, where mutation entails adding normally distributed random vectors to search points generated by recombining parental information. While for simple problems good performance can sometimes be achieved with isotropically distributed mutations, other problems require that mutation vectors be drawn from more general normal distributions. For example, Whitley et al. [2] discuss the use of general normal mutation distributions for tracking ridges. Similarly, on convex quadratic functions it can be observed that isotropic mutations can be increasingly inadequate the more the scales of the axes differ.

For most objective functions, the mutation covariance matrix needs to be adapted continually during optimisation. Rudolph [3] asserts that ideally, that matrix should be proportional to the inverse of the Hessian matrix of the objective function at the current location in search space. As that location changes, so does the local Hessian. Several approaches have been proposed for adapting the covariance matrix of the mutation distribution. Historically, the first is due to Schwefel [4] and represents a straightforward generalisation of mutative self-adaptation for a single step length. The idea is to make the mutation covariance matrix (or, equivalently, information that it can be derived from) part of the genetic makeup of candidate solutions, and to subject it to variation and selection along with the candidate solutions' object components. However, experimental research [3] has shown that very large populations (of a size that is at least quadratic in the dimensionality of the search space) are required for successful adaptation, and that even when large populations are used performance is not necessarily satisfactory.

A more recent approach to adapting the mutation covariance matrix is the covariance matrix adaptation evolution strategy (CMA-ES) proposed by Hansen and Ostermeier [5], [6], [7]. The underlying idea is to gather information about successful search steps, and to use that information to modify the covariance matrix of the mutation distribution in a goal directed, derandomised fashion. Changes to the covariance matrix are such that variances in directions of the search space that have previously been successful are increased while those in other directions passively decrease. The accumulation of information over a number of search steps makes it possible to reliably adapt the covariance matrix even when using small populations. An experimental comparison involving a standard suite of test functions in [7] confirms the superiority of the derandomised approach over the purely mutative one.

Müller et al. [8] and Hansen et al. [9] propose a modification to the update rule for the covariance matrix and show that it scales more favourably with the population size than the original CMA-ES. By using the information present in large populations more effectively, faster adaptation of the covariance matrix can be achieved. Kern et al. [10] and Hansen and Kern [11] experimentally compare the performance of the modified CMA-ES with those of several other evolutionary and estimation of distribution algorithms, both in unimodal and multimodal settings. They find that while strategies that exploit simplifying features such as separability may have advantages on simple objective functions where those features are present, CMA-ES are generally the most effective strategies on non-separable and badly scaled problems.

Noting that the CMA-ES may require a substantial number of time steps for the adaptation of the covariance matrix, Auger et al. [12] propose a least squares approach to explicitly approximate the objective function's Hessian based on data obtained from a quadratic number of function evaluations. Inverting that Hessian yields a matrix that can be used as mutation covariance matrix. The least squares approximation is recomputed periodically, and adaptation in between uses the derandomised CMA approach. On some functions, inverting an explicit approximation to the Hessian yields good mutation covariance matrices very quickly. However,

The authors are with the Faculty of Computer Science, Dalhousie University, Halifax, NS, Canada B3H 1W5 (email: {grahame|dirk}@cs.dal.ca)

while efficient in terms of the number of objective function evaluations required, the approach incurs computational costs of order n^6 for solving the least squares problem. Moreover, it represents a significant departure from the paradigm of evolutionary computation, and more research will need to be done to confirm that it holds promise for the difficult optimisation problems that evolutionary approaches are typically applied to. For example, it is unclear to what degree the least squares approach is affected by discontinuities, noise, etc. (this arguably applies to other covariance matrix adaptation strategies as well).

This paper proposes a novel approach to speeding up the adaptation of the covariance matrix in the CMA-ES. The update rule in the original algorithm is such that the information stored in the covariance matrix decays at a constant rate, and that information from successful steps of the strategy is used to increase variances in directions of the search space that have proven beneficial in the past. Future offspring candidate solutions are thus generated preferably in directions that have proven worth exploring. We propose to make a modification to the update rule for the covariance matrix to the effect that previous directions that have proven especially *unsuccessful* are discouraged in the future. Variances in directions of unsuccessful mutations are actively reduced, thus allowing the strategy to more quickly focus on useful directions. As variances are actively decreased rather than decaying over time, we refer to the resulting strategy as Active-CMA-ES. It will be seen below that substantial benefits can result from active covariance matrix adaptation.

The remainder of this paper is organised as follows. Section II describes evolution strategies with derandomised covariance matrix adaptation as introduced in [7], [8], [9]. In Section III, the modification to that algorithm that results in active covariance matrix adaptation is described. Section IV reports results from computational experiments used to compare the performance of Active-CMA-ES with that of existing strategy variants. Section V concludes with a brief summary and suggestions for future work.

II. Covariance Matrix Adaptation Evolution Strategies

This section provides a brief description of the $(\mu/\mu,\lambda)$-CMA-ES as introduced by Hansen and Ostermeier [7]. The algorithm outlined here is identical to that described by Hansen et al. [9]. In particular, it contains the modifications that lead to improved scaling properties for large populations. In contrast to [7], it does not make use of weighted recombination (that, according to [10], typically leads to only a slight increase in performance).

The $(\mu/\mu,\lambda)$-CMA-ES employs a number of variables that are updated in every iteration of the strategy. Specifically, the search point $\mathbf{x} \in I\!\!R^N$ represents the centroid of the population of candidate solutions. Mutation strength σ and $n \times n$ covariance matrix $\mathbf{C}$ determine the distribution of offspring candidate solutions. Search paths $\mathbf{p}_\sigma$ and $\mathbf{p}_\mathbf{C}$ are two n-dimensional vectors that are used to accumulate information about recent steps of the strategy. The search paths are initialised to be zero; the initial covariance matrix is the unity matrix. The initialisation of the search point and the mutation strength are problem dependent. An iteration of the $(\mu/\mu,\lambda)$-CMA-ES updates those variables using the following seven steps (using "←" to denote the assignment operator):

1) Compute an eigen decomposition $\mathbf{C} = \mathbf{BD}(\mathbf{BD})^\mathrm{T}$ of the mutation covariance matrix such that the columns of $n \times n$ matrix $\mathbf{B}$ are the normalised eigenvectors of $\mathbf{C}$, and $\mathbf{D}$ is a diagonal $n \times n$ matrix the diagonal elements of which are the square roots of the eigenvalues of $\mathbf{C}$.
2) Generate λ offspring candidate solutions
$$\mathbf{y}_i = \mathbf{x} + \sigma \mathbf{BD}\mathbf{z}_i, \qquad i = 1,\ldots,\lambda,$$
where the $\mathbf{z}_i$ are mutation vectors consisting of n independent, standard normally distributed components.
3) Determine the objective function values $f(\mathbf{y}_i)$ and order the offspring candidate solutions according to those values such that $k;\lambda$ denotes the index of the kth best. Compute the average
$$\mathbf{z}^{(\mathrm{avg})} = \frac{1}{\mu}\sum_{k=1}^{\mu} \mathbf{z}_{k;\lambda} \tag{1}$$
of the μ best mutation vectors. Throughout this paper, $\mu = \lambda/4$.
4) Update the search point according to
$$\mathbf{x} \leftarrow \mathbf{x} + \sigma \mathbf{BD}\mathbf{z}^{(\mathrm{avg})}.$$
5) Update the search paths according to
$$\mathbf{p}_\mathbf{C} \leftarrow (1 - c_\mathbf{C})\mathbf{p}_\mathbf{C} + \sqrt{\mu c_\mathbf{C}(2 - c_\mathbf{C})}\mathbf{BD}\mathbf{z}^{(\mathrm{avg})}$$
and
$$\mathbf{p}_\sigma \leftarrow (1 - c_\sigma)\mathbf{p}_\sigma + \sqrt{\mu c_\sigma(2 - c_\sigma)}\mathbf{B}\mathbf{z}^{(\mathrm{avg})},$$
where $c_\mathbf{C} = c_\sigma = 4/(n+4)$.
6) Update the covariance matrix according to
$$\begin{aligned}\mathbf{C} \leftarrow\; & (1 - c_\mathrm{cov})\mathbf{C} \\ & + c_\mathrm{cov}\left(\alpha_\mathrm{cov}\mathbf{p}_\mathbf{C}\mathbf{p}_\mathbf{C}^\mathrm{T} + (1 - \alpha_\mathrm{cov})\mathbf{Z}\right),\end{aligned} \tag{2}$$
where
$$\mathbf{Z} = \mathbf{BD}\left(\frac{1}{\mu}\sum_{k=1}^{\mu}\mathbf{z}_{k;\lambda}\mathbf{z}_{k;\lambda}^\mathrm{T}\right)(\mathbf{BD})^\mathrm{T} \tag{3}$$
and where
$$\begin{aligned}c_\mathrm{cov} =\; & \alpha_\mathrm{cov}\frac{2}{(n+\sqrt{2})^2} \\ & + (1 - \alpha_\mathrm{cov})\min\left(1, \frac{2\mu - 1}{(n+2)^2 + \mu}\right)\end{aligned} \tag{4}$$
with $\alpha_\mathrm{cov} \in [0, 1]$.
7) Update the mutation strength according to
$$\sigma \leftarrow \sigma \cdot \exp\left(\frac{\|\mathbf{p}_\sigma\| - \chi_n}{d_\sigma \chi_n}\right),$$

where $\chi_n = \sqrt{n}(1 - 1/(4n) + 1/(21n^2))$ approximates the expectation of the length of a random vector with independent, standard normally distributed components, and where $d_\sigma = 1 + 1/c_\sigma$ serves as a damping factor.

See [9] for a thorough motivation of the algorithm and the choice of parameter settings. Notice that even though matrix $\mathbf{C}$ is referred to as covariance matrix, the true covariance matrix used for generating offspring candidate solutions is $\sigma^2\mathbf{C}$. Mutation strength σ is dealt with separately from matrix $\mathbf{C}$ in order to be able to adapt the overall step length on a time scale different than that used for adapting the shape of the distribution. Adaptation of the former uses the idea of cumulative step length adaptation introduced by Ostermeier et al. [13]. Adaptation of the latter is done with the implicit goal of increasing the probability of replicating successful steps. The update rule in Eq. (2) multiplies covariance matrix $\mathbf{C}$ with a factor less than 1 in order for old information to slowly decay. Then, two terms are scaled appropriately and added in order to reinforce variances in successful directions. The first of the two terms (that involves $\mathbf{p_C}\mathbf{p_C}^{\mathrm{T}}$) is a matrix of rank 1 that is referred to as the path based term. It relies on the search path $\mathbf{p_C}$ that holds a record of previously successful steps. The second term (that involves matrix $\mathbf{Z}$) is referred to as the population based term and consists of a rank μ matrix. While the path based term contains information about the progress of the population centroid accumulated over the generations, the population based term contains information about individual mutation vectors from the current generation only. The parameter α_{cov} dictates the relative contributions of the path and population based terms. For $\alpha_{\text{cov}} = 1$, the population based term does not contribute to the covariance matrix update at all and the strategy is referred to as Original-CMA-ES (as the population based term was not present when the CMA-ES was first introduced). In [9], Hansen et al. recommend a setting of $\alpha_{\text{cov}} = 1/\mu$ and refer to the resulting strategy as Hybrid-CMA-ES (as both the path and population based terms are used). Notice that according to Eq. (4), for large populations Hybrid-CMA-ES uses a larger value for c_{cov} than Original-CMA-ES, thus enabling faster adaptation.

III. Active Covariance Matrix Adaptation

It has been seen in the past that using information about bad mutations can help speed up the progress of evolution strategies. Weighted recombination refers to the idea of replacing the computation of the average in Eq. (1) with a weighted sum, where the weight of a mutation vector depends on the respective offspring candidate solution's rank in the set of all offspring generated. Better offspring candidate solutions receive larger weights than not as good ones. Hansen and Ostermeier [7] routinely use weighted recombination in the CMA-ES. Rudolph [14] has shown that associating *negative* weights with unfavourable candidate solutions can sometimes lead to even faster progress. In [15] is has been seen that the use of optimal weights can lead to a speed-up by a factor of two and a half compared to the $(\mu/\mu, \lambda)$-ES for the infinite-dimensional sphere model. However, optimal weights differ from objective function to objective function. Moreover, the use of negative weights may lead to unstable behaviour on functions such as f_{discus} defined below, and more work will need to be done in order to better understand the interplay between weighted recombination and step length adaptation. In order to avoid effects resulting from the use of weighted recombination to interfere with our results it is not considered here.

This paper proposes to exploit information about bad mutations not directly for the purpose of accelerating progress, but instead for speeding up the adaptation of the covariance matrix (which, in turn, will lead to faster progress). According to Eqs. (2) and (3), the population based term in the covariance matrix update rule of Hybrid-CMA-ES uses the mutation vectors of the μ best candidate solutions of a generation in order to increase variances in the corresponding directions in search space. In contrast, variances in unfavourable directions that consistently yield bad offspring candidate solutions are decreased merely passively as a result of the multiplication of the covariance matrix with the factor $(1 - c_{\text{cov}})$ in every time step. Active-CMA-ES actively decreases variances in unfavourable directions of the search space by replacing Eq. (3) with

$$\mathbf{Z} = \mathbf{BD}\left(\frac{1}{\mu}\sum_{k=1}^{\mu}\mathbf{z}_{k;\lambda}\mathbf{z}_{k;\lambda}^{\mathrm{T}} - \frac{1}{\mu}\sum_{k=\lambda-\mu+1}^{\lambda}\mathbf{z}_{k;\lambda}\mathbf{z}_{k;\lambda}^{\mathrm{T}}\right)(\mathbf{BD})^{\mathrm{T}}. \quad (5)$$

That is, not only the mutation vectors corresponding to the μ best offspring candidate solutions, but also those corresponding to the μ worst are used. The μ best mutation vectors enter the summation with positive weights; the terms involving the μ worst mutation vectors are given negative weights of the same magnitude.

The coefficients in Eq. (2) have been chosen such that the expectation of the covariance matrix is stationary in the case that selection is random (i.e., that the indices $k;\lambda$ are drawn randomly from $\{1, \ldots, \lambda\}$, as is the case in flat fitness landscapes where $f(\mathbf{y}) = \text{const}$). The expectation of matrix $\mathbf{Z}$ as defined in Eq. (3) has in [9] been seen to equal $\mathbf{C}$ in case of random selection. In contrast, the expectation of matrix $\mathbf{Z}$ defined in Eq. (5) can easily be seen to be zero in that case (see [16]). Thus, in order to maintain stationarity of the expectation of the covariance matrix under random selection, Active-CMA-ES replaces Eq. (2) with

$$\mathbf{C} \leftarrow (1 - c_{\text{cov}})\mathbf{C} + c_{\text{cov}}\mathbf{p_C}\mathbf{p_C}^{\mathrm{T}} + \beta\mathbf{Z}. \quad (6)$$

Finally, we have replaced the definition of the cumulation parameter in Eq. (4) with

$$c_{\text{cov}} = \frac{2}{(n + \sqrt{2})^2}.$$

This is the same value that is used in Original-CMA-ES. Notice that no significant new computational costs are incurred as a result of the modifications.

It remains to determine appropriate values for the new parameter β in Eq. (6) that dictates the contribution of matrix $\mathbf{Z}$ in the covariance matrix adaptation process. In general, larger values of β allow for faster adaptation, albeit at the cost of a loss of stability of the adaptation process. If β is chosen too large, then fluctuations of the values of the covariance matrix occur that result in reduced performance. In addition, too large values of β may lead to the loss of positive definiteness of the covariance matrix. As the eigenvalues of $\mathbf{C}$ are computed after every update, a loss of positive definiteness is easily detected and can be fixed. However, it is undesirable as it shows that covariance matrix adaptation does not work as envisioned.

Appropriate values for β have been determined in a process closely akin to that used in [9] to obtain the expression given in Eq. (4) for c_{cov}. A number of computational experiments were performed in which the number of generations required to reach an objective function value f_{stop} on the ellipsoid function $f_{\text{ellipsoid}}$ was measured for search space dimensionalities $n \in \{2, 3, 5, 10, 20, 40, 80\}$ and population sizes $\mu \in \{2, n\}$. See Section IV for a definition of that function and initialisation conditions, and see [16] for a more detailed description of the process. The choice of population sizes was made in order to be able to achieve good performance for both small ($\mu = 2$) and large ($\mu = n$) populations. For every combination of n and μ considered, the value of β that gave the best performance (i.e., that resulted in the minimum number of generations required to reach f_{stop}) while maintaining positive definiteness of the covariance matrix was recorded. Similar to [9], those optimal values of β were then multiplied with a factor of 0.6, resulting in a more conservative choice. This suboptimal choice of β leads to a loss in performance in the order of magnitude of 20%, but has the advantage of improving the robustness of the adaptation process. Then, a function of the form

$$\beta(n, \mu) = \frac{c_1\mu - c_2}{(n + c_3)^2 + c_4\mu}$$

was fitted to the conservative β values using non-linear least-squares regression. The choice of function type was motivated by the fact that the same functional form was used in [9] for the cumulation parameter c_{cov}. The values of the parameters obtained by performing least-squares regression (rounded to integers for notational convenience) are $c_1 = 4$, $c_2 = 2$, $c_3 = 12$, and $c_4 = 4$, resulting in

$$\beta = \frac{4\mu - 2}{(n + 12)^2 + 4\mu} \tag{7}$$

as an expression for the parameter β. While there is no guarantee that this experimentally derived expression ensures that the covariance matrix will remain positive definite throughout, a loss of positive definiteness has not been observed in any of the experiments described in Section IV for any of the test functions considered.

TABLE I

TEST FUNCTIONS AND STOPPING CRITERIA.

name	function	f_{stop}
sphere	$f_{\text{sphere}}(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$	10^{-10}
ellipsoid	$f_{\text{ellipsoid}}(\mathbf{x}) = \sum_{i=1}^{n} a^{\frac{i-1}{n-1}} x_i^2$	10^{-10}
cigar	$f_{\text{cigar}}(\mathbf{x}) = x_1^2 + \sum_{i=2}^{n} a x_i^2$	10^{-10}
discus	$f_{\text{discus}}(\mathbf{x}) = a x_1^2 + \sum_{i=2}^{n} x_i^2$	10^{-10}
cigar-discus	$f_{\text{cigdis}}(\mathbf{x}) = a x_1^2 + \sum_{i=2}^{n-1} a^{\frac{1}{2}} x_i^2 + x_n^2$	10^{-10}
parab. ridge	$f_{\text{parabR}}(\mathbf{x}) = -x_1 + 100 \sum_{i=2}^{n} x_i^2$	-10^{10}
two-axes	$f_{\text{twoaxes}}(\mathbf{x}) = \sum_{i=1}^{\lfloor n/2 \rfloor} a x_i^2 + \sum_{i=\lfloor n/2+1 \rfloor}^{n} x_i^2$	10^{-10}
diff. powers	$f_{\text{diffpow}}(\mathbf{x}) = \sum_{i=1}^{n} \lvert x_i \rvert^{2+10\frac{i-1}{n-1}}$	10^{-15}
Rosenbrock	$f_{\text{Rosen}}(\mathbf{x}) = \sum_{i=1}^{n-1} \Big(100\big(x_i^2 - x_{i+1}\big)^2 + (x_i - 1)^2\Big)$	10^{-10}

IV. Experimental Evaluation

This section compares the performance of Active-CMA-ES with those of Original-CMA-ES and Hybrid-CMA-ES. The methodology of the comparison is the same as in [7], [9]. Section IV-A briefly describes the test environments; Section IV-B reports and discusses results from experiments in those environments.

A. Experimental Setup

The test functions considered in the comparison are listed in Table I and are commonly used in related work [7], [9], [12]. In all experiments described below the number of generations required to reach an objective function value of f_{stop} is used as a performance measure, where the respective values of f_{stop} are also given in the table. Initialisation conditions are the same as in [9] (i.e., $\mathbf{x}$ is initialised by setting all components to 1 and the initial mutation strength is 1 except for Rosenbrock's function where the components of $\mathbf{x}$ are initialised to zero and the initial mutation strength is 0.1). The scaling coefficient a that appears in the definitions of $f_{\text{ellipsoid}}$, f_{cigar}, f_{discus}, and f_{cigdis} is 10^6 except when noted otherwise. Furthermore, as in [9], the number of offspring candidate solutions generated per time step is $\lambda = 4\mu$.

All of the functions considered in this paper are unimodal. Moreover, the majority of them are separable in that they could be minimised effectively in a sequence of independent minimisations with respect to single variables, while the $n - 1$ other variables are kept fixed. However, as noted by Hansen [9], CMA-ES do not exploit separability of the objective, and their performance is invariant under rotations of the search space. All experiments could have been (and indeed have been) performed on variables $\mathbf{y}$ obtained by multiplying $\mathbf{x}$ with a random orthonormal base, making all functions but f_{sphere} non-separable. The random orthonormal base has been generated as described in [7]. All experiments

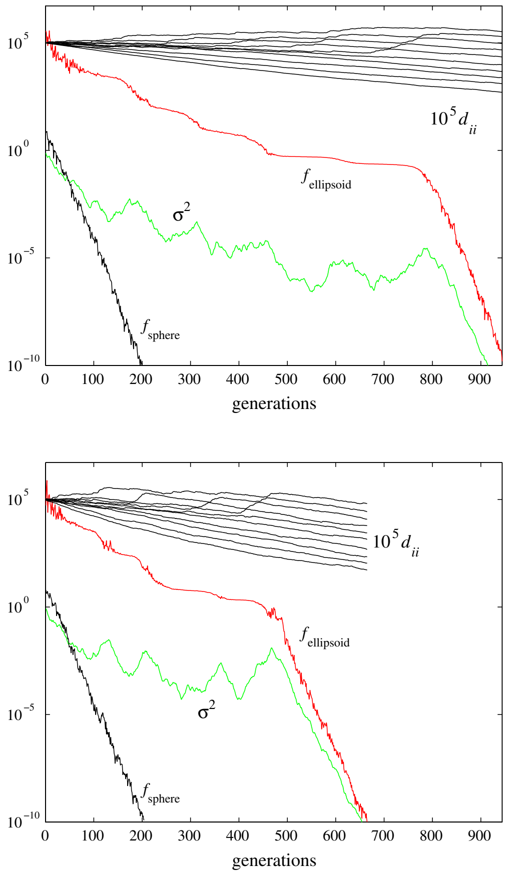

Fig. 1. Single runs of Hybrid-CMA-ES (top) and Active-CMA-ES (bottom) with $\mu = 2$ on the ellipsoid function $f_{\text{ellipsoid}}$ with $n = 10$. See the text for an explanation of the data shown.

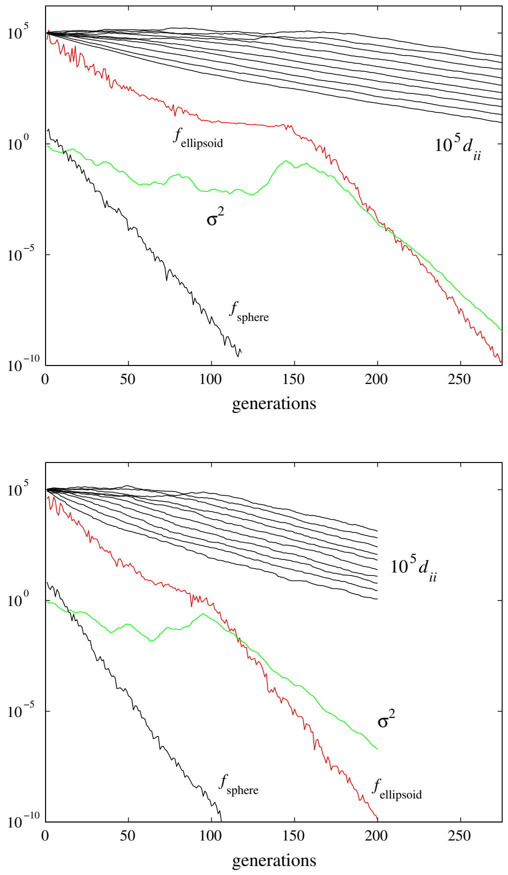

Fig. 2. Single runs of Hybrid-CMA-ES (top) and Active-CMA-ES (bottom) with $\mu = 10$ on the ellipsoid function $f_{\text{ellipsoid}}$ with $n = 10$. See the text for an explanation of the data shown.

were performed in MATLAB, using a modified version of the CMA-ES source code provided by Hansen et al. [9]. The exploration of multimodal functions remains as a task for future work.

B. Results

Typical single runs of Hybrid-CMA-ES and Active-CMA-ES on the ellipsoid function $f_{\text{ellipsoid}}$ with $n = 10$ are shown in Figure 1 (small population; $\mu = 2$) and Figure 2 (large population; $\mu = n$). Each graph contains objective function values $f(\mathbf{x})$, the squared mutation strength σ^2, and the sorted principal axis lengths of the mutation distribution (diagonal elements of matrix $\mathbf{D}$, scaled with 10^5 to achieve readability) plotted against the number of generations. Also shown for comparison is each strategy's performance on the sphere function f_{sphere}. For optimally adapted covariance matrix (i.e., for $\mathbf{C}$ proportional to the inverse Hessian), progress on any convex quadratic function is as fast as on the sphere, and the slopes of the respective $f(\mathbf{x})$ curves agree. This can be observed in Figures 1 and 2 in the final phase of the runs, providing evidence that covariance matrix adaptation is indeed successful in generating near optimal covariance matrices. It can also be seen that Active-CMA-ES reaches f_{stop} faster than Hybrid-CMA-ES, especially for $\mu = 2$. The majority of the speed-up occurs in the phase of optimisation where the algorithm adapts its covariance matrix while progress in terms of the improvement of functions values is relatively slow.

Figure 3 contrasts the performance of Original-CMA-ES, Hybrid-CMA-ES, and Active-CMA-ES across all of the test functions from Table I for $n = 10$ and both small and large populations. Original-CMA-ES has been included in the comparison as it performs better than Hybrid-CMA-ES on some of the test functions if small populations are used. Ten runs were performed for each objective function and value of μ until a function value of f_{stop} or less was achieved. The data shown in the figure are median values of the number of generations required. Not shown, the standard deviations from the mean of the measurements are in most cases smaller than the differences between values measured for different strategies. It can be seen from the figure that Active-CMA-ES in all cases reaches f_{stop} as fast as or faster than both Original-CMA-ES and Hybrid-CMA-ES. The performance advantage of Active-CMA-ES reaches from none on the sphere function (where no covariance matrix adaptation is required) and $\mu = 2$ to over 40% compared to the performance of Hybrid-CMA-ES on f_{diffpow} and $\mu = n$. Overall,

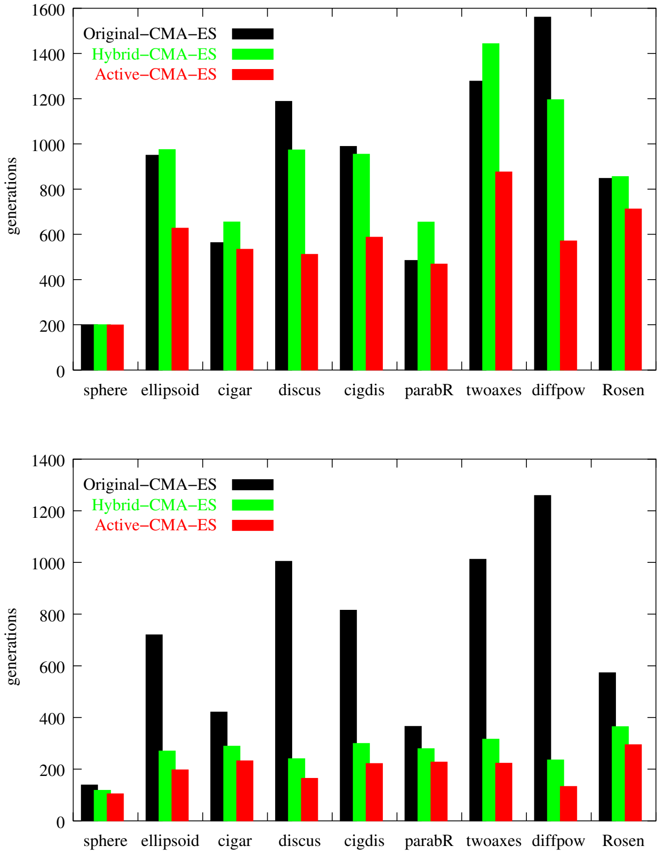

Fig. 3. Number of generations required to reach f_{stop} for the test functions in Table I with $n = 10$. Each bar shows the median value from ten independent runs of the respective strategy variant for $\mu = 2$ (top) and $\mu = n$ (bottom).

the performance of Active-CMA-ES is particularly good on functions with an eigenvalue spectrum of the Hessian that is dominated by one value much larger than the others, such as f_{discus}. On such functions, the variance of the mutation distribution in the corresponding direction in search space must be small in order to make measurable progress. Active-CMA-ES's ability to actively reduce variances is helpful for that purpose.

Figures 4, 5, and 6 illustrate how the performances of the three CMA-ES variants scale with the search space dimensionality n for $f_{\text{ellipsoid}}$, f_{cigar}, and f_{discus}. The corresponding plots for the test functions not shown here look qualitatively similar. Simulations have been run for both small ($\mu = 2$) and large ($\mu = n$) populations and with $n \in \{2, 3, 5, 10, 20, 40, 80\}$. Again, ten runs were performed for each case. Median values of the number of generations required to reach f_{stop} are plotted on a log-log scale, with errorbars indicating the standard deviation from the mean. Straight lines in log-log plots indicate a power law relationship of the form $y = z \cdot x^b$, where b corresponds to the slope of the straight line. To quickly identify the approximate scaling complexity of the strategies, each figure contains sloping dotted lines representing scaling complexities of n and n^2.

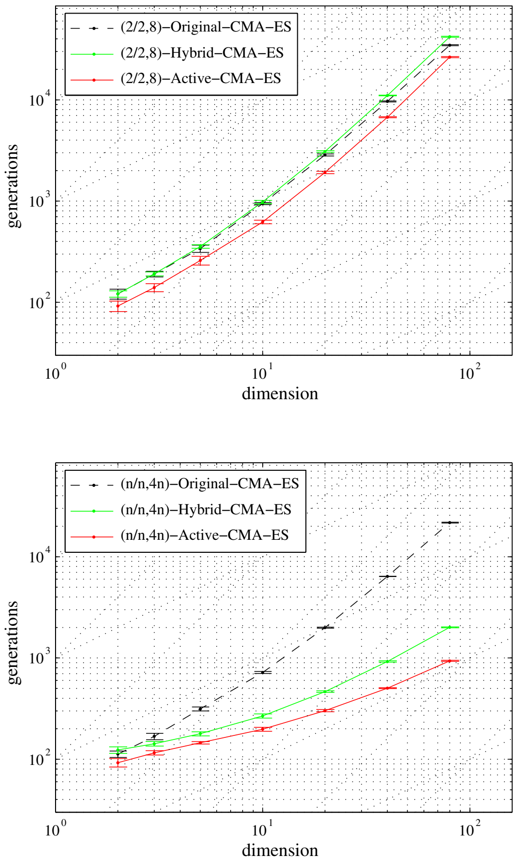

Fig. 4. Performance of Active-CMA-ES compared with those of Original-CMA-ES and Hybrid-CMA-ES. Shown is the number of generations required to reach f_{stop} on $f_{\text{ellipsoid}}$ for $\mu = 2$ (top) and $\mu = n$ (bottom) plotted against the search space dimensionality n.

Overall, it can be seen from the figures that the performance advantages of Active-CMA-ES that have been observed for $n = 10$ are across the entire range of search space dimensionalities. For the ellipsoid function $f_{\text{ellipsoid}}$, the performance advantage of Active-CMA-ES over Hybrid-CMA-ES grows to about 50% for $\mu = n$ and $n = 80$. For the cigar function f_{cigar}, the performance advantage is relatively independent of the search space dimensionality and in the vicinity of 20%. The benefits of active covariance matrix adaptation are most pronounced on functions such as the discus function f_{discus}, where the advantage of Active-CMA-ES grows significantly with increasing n. For small populations and $n = 80$, Active-CMA-ES requires about 75% fewer generations to reach f_{stop} than either Original-CMA-ES or Hybrid-CMA-ES. For large populations, the advantage over Hybrid-CMA-ES is about 45%. From the slope of the corresponding line in the upper subfigure of Figure 6, it appears that for a wide range of search space dimensionalities the number of generations required to reach f_{stop} on the discus function scales closer to linearly than to quadratically even if small population are used.

Also of interest is the scaling behaviour of the strategies

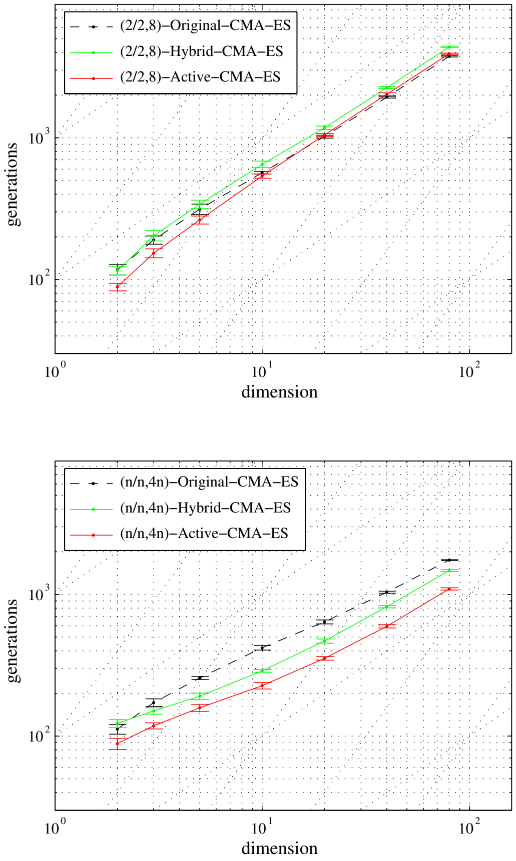

Fig. 5. Performance of Active-CMA-ES compared with those of Original-CMA-ES and Hybrid-CMA-ES. Shown is the number of generations required to reach f_{stop} on f_{cigar} for $\mu = 2$ (top) and $\mu = n$ (bottom) plotted against the search space dimensionality n.

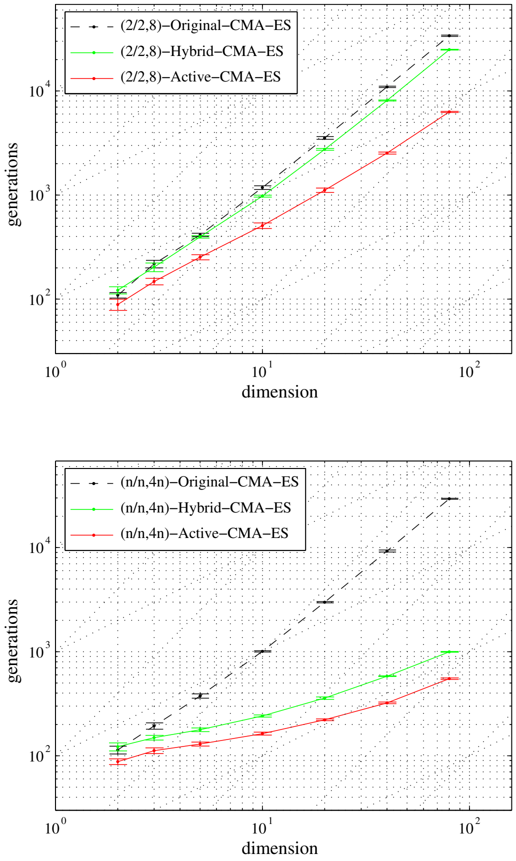

Fig. 6. Performance of Active-CMA-ES compared with those of Original-CMA-ES and Hybrid-CMA-ES. Shown is the number of generations required to reach f_{stop} on f_{discus} for $\mu = 2$ (top) and $\mu = n$ (bottom) plotted against the search space dimensionality n.

with regard to the coefficient a that determines the relative length of the axes of $f_{\text{ellipsoid}}$, f_{cigar}, f_{discus}, and f_{cigdis}. With increasing values of a, the axis misscaling becomes more pronounced, and the strategies require more generations to generate good covariance matrices. Figure 7 illustrates how the degree of axis misscaling affects the three CMA-ES variants for the case of the ellipsoid function $f_{\text{ellipsoid}}$ and $n = 20$. Results for the other test functions are similar and can be found in [16]. It can be seen from the figure that the advantage of active covariance matrix adaptation increases with an increasing degree of axis misscaling as the gap between the three strategy variants widens.

Finally, not shown in this paper, additional experiments have been performed on scaled versions of Rastrigin's function as defined in [7]. See [16] for details. That function combines axes of widely differing lengths with multimodality as an additional complication. It appears from those experiments that the global search performance of the strategy is not compromised by active covariance matrix adaptation. Hybrid-CMA-ES and Active-CMA-ES perform very similarly. However, a more careful experimental investigation along the lines of [11] will be necessary in order to confirm those results.

V. Summary and Conclusions

To conclude, this paper has introduced a modification to the CMA-ES algorithm. The objective for the modification was to let not only the best, but also information about the worst offspring candidate solutions contribute to the adaptation of the covariance matrix. Rather than waiting for variances in unpromising directions in search space to passively decay, the modified strategy actively reduces them and is therefore referred to as Active-CMA-ES. Care has to be taken to ensure that the mutation covariance matrix remains positive definite. The values for the parameter β that have been obtained from experiments on the ellipsoid function have proven useful in that in none of our experiments on any the objective functions considered, a loss of positive definiteness has been observed. An experimental comparison on a standard set of unimodal test functions has seen Active-CMA-ES outperform both Original-CMA-ES and Hybrid-CMA-ES. The advantages of active covariance matrix adaptation are most pronounced on objective functions with eigenvalue spectra of their Hessians that are dominated by one value that is much larger than the others, such as the

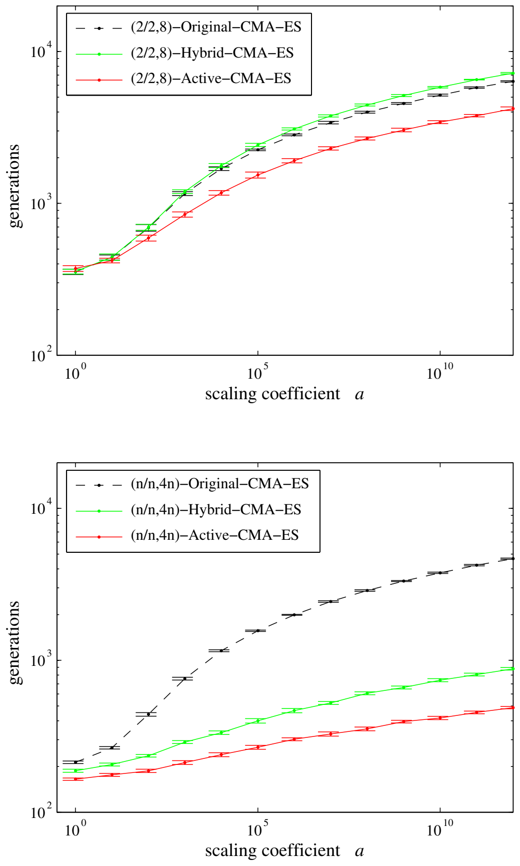

Fig. 7. Performance of Active-CMA-ES compared with those of Original-CMA-ES and Hybrid-CMA-ES. Shown is the number of generations required to reach f_{stop} on $f_{\text{ellipsoid}}$ with $n = 20$ for $\mu = 2$ (top) and $\mu = n$ (bottom) plotted against the coefficient a that determines the axis scaling.

discus function f_{discus}. For those functions, the variance of the mutation distribution in that direction can be reduced much faster using active covariance matrix adaptation. However, sizable benefits can also be observed for other objective functions the eigenvalue spectrum of which is more evenly distributed, such as $f_{\text{ellipsoid}}$ and f_{Rosen}. For $n = 10$, the performance advantage of Active-CMA-ES ranges from none (on the sphere, where no covariance matrix adaptation is required) to more than 40% (for $\mu = 2$ on f_{diffpow}). Larger performance advantages can be observed on some functions for higher search space dimensionalities as well as in case of larger axis ratios.

In future work, we plan to study the use of weighted recombination in Active-CMA-ES. Weighted recombination has been found to yield a (moderate) performance increase for both Original-CMA-ES and Hybrid-CMA-ES, and we expect a similar increase for Active-CMA-ES. A further subject of future investigation is the performance of Active-CMA-ES on multimodal test functions. While preliminary experiments have provided no evidence of a decrease in global search performance as a result of active covariance matrix adaptation, a more careful study along the lines of [11] will be necessary to confirm this.

Acknowledgement

This research was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

[1] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies — A comprehensive introduction," *Natural Computing*, vol. 1, no. 1, pp. 3–52, 2002.

[2] D. Whitley, M. Lunacek, and J. Knight, "Ruffled by ridges: How evolutionary algorithms can fail," in *Genetic and Evolutionary Computation — GECCO 2004*, K. Deb, R. Poli, W. Banzhaf, H.-G. Beyer, E. Burke, P. Darwen, D. Dasgupta, D. Floreano, J. Foster, M. Harman, O. Holland, P. L. Lanzi, L. Spector, A. Tettamanzi, D. Thierens, and A. Tyrell, Eds. Springer Verlag, Heidelberg, 2004, pp. 294–306.

[3] G. Rudolph, "On correlated mutations in evolution strategies," in *Parallel Problem Solving from Nature — PPSN II*, R. Männer and B. Manderick, Eds. Elsevier, Amsterdam, 1992, pp. 105–114.

[4] H.-P. Schwefel, *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie*. Birkhäuser Verlag, Basel, 1977.

[5] N. Hansen and A. Ostermeier, "Adapting arbitrary normal distributions in evolution strategies: The covariance matrix adaptation," in *Proceedings of the Third IEEE Conference on Evolutionary Computation*. IEEE Press, Piscataway, NJ, 1996, pp. 312–317.

[6] N. Hansen, *Verallgemeinerte individuelle Schrittweitenregelung in der Evolutionsstrategie*. Mensch & Buch Verlag, Berlin, 1998.

[7] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies," *Evolutionary Computation*, vol. 9, no. 2, pp. 159–195, 2001.

[8] S. D. Müller, N. Hansen, and P. Koumoutsakos, "Increasing the serial and the parallel performance of the CMA-evolution strategy with large populations," in *Parallel Problem Solving from Nature — PPSN VII*, J. J. M. Guervós, P. Adamidis, H.-G. Beyer, J.-L. Fernández-Villacañas, and H.-P. Schwefel, Eds. Springer Verlag, Heidelberg, 2002, pp. 422–431.

[9] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evolutionary Computation*, vol. 11, no. 1, pp. 1–18, 2003.

[10] S. Kern, S. D. Müller, N. Hansen, D. Büche, J. Ocenasek, and P. Koumoutsakos, "Learning probability distributions in continuous evolutionary algorithms — A comparative review," *Natural Computing*, vol. 3, no. 1, pp. 77–112, 2004.

[11] N. Hansen and S. Kern, "Evaluating the CMA evolution strategy on multimodal test functions," in *Parallel Problem Solving from Nature — PPSN VIII*, X. Yao, E. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, Eds. Springer Verlag, Heidelberg, 2004, pp. 282–291.

[12] A. Auger, M. Schoenauer, and N. Vanhaecke, "LS-CMA-ES: A second-order algorithm for covariance matrix adaptaion," in *Parallel Problem Solving from Nature — PPSN VIII*, X. Yao, E. Burke, J. A. Lozano, J. Smith, J. J. Merelo-Guervós, J. A. Bullinaria, J. Rowe, P. Tiňo, A. Kabán, and H.-P. Schwefel, Eds. Springer Verlag, Heidelberg, 2004, pp. 182–191.

[13] A. Ostermeier, A. Gawelczyk, and N. Hansen, "Step-size adaptation based on non-local use of selection information," in *Parallel Problem Solving from Nature — PPSN III*, Y. Davidor, H.-P. Schwefel, and R. Männer, Eds. Springer Verlag, Heidelberg, 1994, pp. 189–198.

[14] G. Rudolph, *Convergence Properties of Evolutionary Algorithms*. Verlag Dr. Kovač, Hamburg, 1997.

[15] D. V. Arnold, "Optimal weighted recombination," in *Foundations of Genetic Algorithms 8*, A. H. Wright, M. D. Vose, K. A. De Jong, and L. M. Schmitt, Eds. Springer Verlag, Heidelberg, 2005, pp. 215–237.

[16] G. A. Jastrebski, "Improving evolution strategies through actice covariance matrix adaptation," Master's thesis, Faculty of Computer Science, Dalhousie University, 2005.