Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 246 (2024) 1389–1398

Procedia Computer Science

www.elsevier.com/locate/procedia

28th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2024)

Adaptability and Efficiency in Population Management: A multi-population CMA-ES Strategy for High-Dimensional Optimization

Bernardo Morales-Castañeda[a], Erick Rodríguez-Esparza[b], Diego Oliva[a,*], Mario A. Navarro[a], Itzel Aranguren[a], Angel Casas-Ordaz[a], Luis A. Beltran[a], Saúl Zapotecas-Martínez[c]

[a]*Universidad de Guadalajara, CUCEI, Guadalajara, Mexico.*
[b]*University of Deusto, Bilbao, Spain.*
[c]*Instituto Nacional de Astrofísica Óptica y Electrónica, INAOE, Puebla, Mexico.*

Abstract

In the context of evolutionary algorithms, having the ability to adapt to any search space within an optimization problem is an essential task. Appropriately adapting the population can lead to better solutions and more efficient use of function call resources. This article presents a renewed approach to population management inspired by modifying the well-known Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm. The proposed strategy aims to improve the algorithm's population adaptability to the search space and optimize function evaluations. Statistically evaluated experimental test outcomes demonstrate significantly better performance on high-dimensional problems in comparison to the original CMA-ES and seven other known evolutionary algorithms in the literature.

Keywords:
Evolutionary algorithms; Population; Covariance matrix; CMA-ES Function evaluation.

1. Introduction

Evolutionary algorithms (EAs) are powerful tools for solving complex optimization problems. They can mimic natural selection processes, which has enabled them to find optimal solutions in search spaces from different domains.

* Corresponding author.
E-mail address: diego.oliva@cucei.udg.mx

10.1016/j.procs.2024.09.579

However, the success of EAs is intrinsically linked to their ability to adapt to the specific characteristics of the problem at hand. Indeed, the operators and restructure of an EA are expected to be flexible and problem-independent [3].

Among a vast range of EAs, the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) stands out for its remarkable performance in high-dimensional optimization problems [8, 10]. CMA-ES employs a covariance matrix to guide the search, dynamically adapting it to the underlying structure of the search space. This adaptation is crucial for efficient exploration and exploitation of the problem landscape. The CMA-ES has been widely applied to solve problems as in optimal control [14], in the tuning of parameters of support vector machines (SVM) [6], and for the optimal design of truss structures with constraints [15], to mention some. As with other EAs, the CMA-ES has some drawbacks that affect its performance. CMA-ES faces challenges regarding memory and computational requirements when applied to large-scale optimization problems. Furthermore, it discards valuable information by focusing solely on the best individuals [11, 2]. In the related literature, finding different variants of the CMA-ES is possible. In [12] proposes a hybridization of the CMA-Es with an improved differential evolution (DE) version. A variant of the CMA-ES that involves local search and anisotropic eigenvalue adaptation was introduced in [13] for solving engineering problems. Also, a restart CMA-ES that increases the size of the population is introduced in [1]. Another recent modification is the cooperative coevolutionary CMA-ES with landscape-aware grouping in noisy environments [18].

This article proposes a novel approach to population management within CMA-ES, called multi-population CMA-ES with stagnation control (MCMAES-SC). The MCMAES-SC further enhances its adaptability and overall performance. In the MCMAES-SC, multiple independent sub-populations are updated by the same amount of covariance matrices. The proposed version of the CMA-ES has a strategy that permits to verify when a sub-population is stagnated. This permits the MCMAES-SC to escape from sub-optimal solutions and provides a better exploration of the search space. To evaluate the effectiveness of our proposed strategy, we conduct extensive experimental comparisons on a range of benchmark functions, including a selection of high-dimensional problems. The experimental results demonstrate that the MCMAES-SC consistently outperforms the original CMA-ES algorithm, achieving significantly better solutions and requiring fewer function evaluations. Moreover, the proposed algorithm is compared with seven other well-known EAs, and observed that it consistently outperforms all of them in terms of optimization performance and efficiency.

The findings highlight the importance of adapting the population management strategy within EAs, particularly in high-dimensional optimization problems. The proposed CMA-ES modification demonstrates a clear advantage over the original algorithm and other established EAs, suggesting promising avenues for further research in EA development. The remainder of the sections are organized as follows: Section 2 describes the concepts of the CMA-ES. In Section 3, the modifications and steps of the MCMAES-SC are described. Section 4 presents the experiments and comparisons. Finally, in Section 5, some conclusions and future works are discussed.

2. Covariance Matrix Adaptation Evolution Strategy (CMA-ES)

The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm [8, 10] stands as a stochastic algorithm designed to optimize a single objective. CMA-ES is recognized as a classic and advanced black-box optimization technique that adjusts the covariance matrix of a multivariate normal search distribution. In the context of minimizing a function $f : \mathbf{R}^n \to \mathbf{R}$, expressed as $\mathbf{x} \mapsto f(\mathbf{x})$, where the goal is to discover one or more search points (candidate solutions), $\mathbf{x} \in \mathbf{R}^n$, with the smallest possible function value $f(x)$, CMA-ES generates a population of new search points (individuals/offsprings) by employing a multivariate normal distribution for sampling. The covariance matrix adaptation is directed at aligning the search distribution with the contour lines of the objective function f that is to be minimized. The fundamental equation for sampling the search points is outlined as follows:

$$\mathbf{x}_k^{(\tau+1)} \sim \mathbf{m}^\tau + \sigma^\tau \times \mathcal{N}(\mathbf{0}, \mathbf{C}^\tau) \; for \; k = 1, \ldots, \lambda \tag{1}$$

where $\mathbf{x}_k^{(\tau+1)}$ is k-th search point (individual/offspring) from generation $\tau + 1$, $\sim$ denotes the same distribution on the left and right side, $\mathbf{m}^\tau$ is the mean value of the search distribution, σ^τ is the "overall" standard deviation or step-size and $\mathcal{N}(\mathbf{0}, \mathbf{C}^\tau)$ is a multivariate normal distribution with zero mean and covariance matrix $\mathbf{C}^\tau$, at generation τ.

The new mean vector $\mathbf{m}^{\tau+1}$ of the search distribution is computed through *weighted recombination* of the μ best candidate solutions, and the final equation rewritten as an update of $\mathbf{m}$ is:

$$\begin{aligned}\mathbf{m}^{\tau+1} &= \sum_{i=1}^{\mu} \omega_i \mathbf{x}_{i:\lambda}^{\tau+1} \\ \mathbf{m}^{\tau+1} &= \mathbf{m}^{\tau} + \sum_{i=1}^{\mu} \omega_i (\mathbf{x}_{i:\lambda}^{\tau+1} - \mathbf{m}^{\tau})\end{aligned} \tag{2}$$

Where the positive weight coefficients for recombination $\sum_{i=1}^{\mu} \omega_i = 1$ with $\omega_i > 0$, are typically chosen linearly on the log scale with $\mu \leq \lambda/2$. To construct the evolution path $\mathbf{p}_c \in \mathbf{R}^n$, starting with $\mathbf{p}_c = 0$, the step size σ is disregarded:

$$\mathbf{p}_c^{\tau+1} = (1 - c_c)\mathbf{p}_c^{\tau} + \sqrt{c_c(2 - c_c)\mu_{\text{eff}}}\,\frac{\mathbf{m}^{\tau+1} - \mathbf{m}^{\tau}}{\sigma_{\tau}} \tag{3}$$

$c_c \leq 1$, and $1/c_c$ is the backward time horizon of the evolution path $\mathbf{p}_c$. The factor $\sqrt{c_c(2 - c_c)\mu_{\text{eff}}}$ is a normalitation constant for $\mathbf{p}_c$. Because, in general, the expected length of the evolution path $\mathbf{p}_c^{\tau+1}$ from Eq. 3 depends on its direction, a conjugate evolution path is constructed, and to control the step-size σ^{τ}, it utilized a sum of successive steps. The method is applied independently of the covariance matrix update and is also called commutative path length control or cumulative step length adaptation (CSA). The evolution path is updated as follows:

$$\mathbf{p}_{\sigma}^{\tau+1} = (1 - c_{\sigma})\mathbf{p}_{\sigma}^{\tau} + \sqrt{c_{\sigma}(2 - c_{\sigma})\mu_{\text{eff}}}\,\mathbf{C}^{\tau^{-1/2}}\frac{\mathbf{m}^{\tau+1} - \mathbf{m}^{\tau}}{\sigma_{\tau}} \tag{4}$$

Where $\mu_{\text{eff}} = \left(\sum_{i=1}^{\mu} \omega_i^2\right)^{-1}$ is the variance effective selection mass, from definition of ω_i its derive $1 \leq \mu_{\text{eff}} \leq \mu$. $\mathbf{p}_{\sigma}^{\tau} \in R_n$ is the conjugate evolution path at generation τ, and $\mathbf{C}^{\tau^{-\frac{1}{2}}} \stackrel{\text{def}}{=} \mathbf{B}^{\tau}\mathbf{D}^{\tau^{-1}}\mathbf{B}^{\tau^{T}}$. Because $\sigma^{\tau} > 0$, equation 4 is equivalent to:

$$\sigma^{\tau+1} = \sigma^{\tau} \times \exp\left(\frac{c_{\sigma}}{d_{\sigma}} \times \left(\frac{||\mathbf{p}_{\sigma}^{\tau+1}||}{E||\mathcal{N}(\mathbf{0}, \mathbf{I})||} - 1\right)\right) \tag{5}$$

Where $E||\mathcal{N}(\mathbf{0}, \mathbf{I})|| = \sqrt{2}\Gamma(\frac{n+1}{2})/\Gamma(\frac{n}{2}) \approx \sqrt{n}(1 - \frac{1}{4n} + \frac{1}{21n^2})$, expectation of the Euclidean norm of a $\mathcal{N}(\mathbf{0}, \mathbf{I})$ distributed random vector. Finally, the covariance matrix is updated by means of *rank-one* and *rank-μ*:

$$\begin{aligned}\mathbf{C}^{\tau+1} &= (1 - c_1 - c_{\mu} \sum \omega_j)\mathbf{C}^{\tau} \\ &+ c_1 \underbrace{\mathbf{p}_c^{(\tau+1)}\mathbf{p}_c^{(\tau+1)^T}}_{\texttt{rank-one update}} + \underbrace{c_{\mu} \sum_{i=1}^{\lambda} \omega_i \mathbf{y}_{\mathbf{i}:\lambda}^{\tau+\mathbf{1}} \left(\mathbf{y}_{\mathbf{i}:\lambda}^{\tau+\mathbf{1}}\right)^T}_{\texttt{rank-}\mu\texttt{ update}}\end{aligned} \tag{6}$$

Where $c_1 \approx 2/n^2$, $c_\mu \approx min(\mu_{eff}/n^2, 1 - c_1)$, $\mathbf{y}^{\tau+1}_{\mathbf{i}:\lambda} = \left(\mathbf{x}^{\tau+1}_{\mathbf{i}:\lambda} - \mathbf{m}^\tau\right)/\sigma^\tau$, and $\sum \omega_j = \sum_{i=1}^{\lambda} \omega_i \approx -c_1/c_u$. For Default Parameters, see Table 1 in Appendix A of the document [9].

3. Multi-population CMA-ES with stagnation control

After thoroughly analyzing the original CMA-ES algorithm, a couple of alterations are proposed. These changes are intended to increase the algorithm's effectiveness in adapting to the search space of any optimization problem and provide more efficient utilization of the allotted functions evaluations. These enhancements are related to how the original CMA-ES algorithm's population is created and controlled. In the original approach, a single population is utilized, with its behavior being influenced by the covariance matrix of its distribution. Throughout multiple iterations, this matrix is fine-tuned to better align with the characteristics of the fitness landscape. This adjustment increases the likelihood of exploring favorable regions within the search space. The covariance matrix is modified by evaluating the distinctions between the chosen solutions and the current mean. This allows the algorithm to estimate parameter correlations and adapt the matrix accordingly [9].

In the proposed MCMAES-SC algorithm, we aim to better adapt the shape of the fitness landscape by employing multiple covariance matrices, governing the evolution of multiple independent sub-populations. In addition, the aforementioned matrices are analyzed through iterations to detect when a specific subgroup has stagnated, which occurs when multiple iterations reveal negligible changes in a specific covariance matrix. In this situation, the stagnated subgroup is eliminated to free the function evaluations employed in each iteration to be utilized by the rest of the population. The flowchart of the proposed methodology is depicted in Figure 1

3.1. Independent covariance matrices for multiple populations

In the proposed methodology, a new parameter *groupsN* is introduced to control the number of sub-populations, which, in turn, determines how many covariance matrices are going to be employed. To this end, the whole population is divided equally between the different sub-populations. The initial population λ and the sub-group population *subPopN* are defined by the following equations:

$$\begin{aligned} \lambda &= 4 + \lfloor 3 \cdot \log Dims \rfloor \cdot 10 \\ subPopN &= \lfloor \lambda / groupsN \rfloor \end{aligned} \tag{7}$$

For each sub-population, independent control parameters (recombination weights, number of effective solutions, step size control parameters, and covariance update parameters) [9] are initialized and updated through the iterations. After the parameters are initialized, the initial points for each sub-population are created and evaluated. To ensure proper distribution of each sub-population in the search space, the initial points are generated utilizing Latin hypercube sampling [17]. Once all control parameters and initial candidate solutions are computed, each sub-population evolves independently, following the same guidelines as in the original CMA-ES algorithm. Figure 2 shows an example of the expected behavior of the MCMAES-SC algorithm.

3.2. Elimination of stagnated subgroups

Ensuring an efficient utilization of the allotted function evaluations is a severely underused strategy to increase the performance of a metaheuristic algorithm. Optimizing the usage of function evaluations can explain this increased performance; it allows the best-performing members of the population to move more frequently, while the members who have become stuck are eliminated so they can't waste function evaluations in future movements. For this strategy to be effective, there must be a way to correctly determine when population members have become stuck and are not contributing to the optimization process.

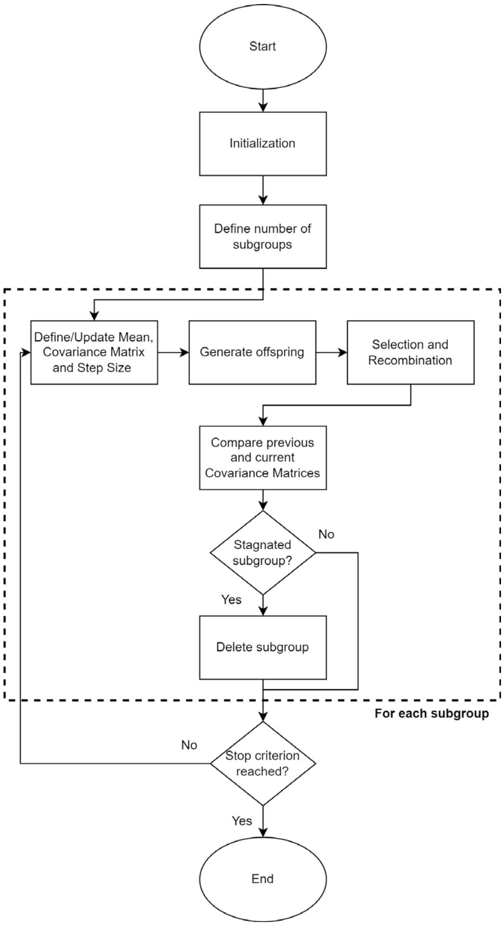


Fig. 1. Flowchart of the MCMAES-SC algorithm.

The MCMAES-SC algorithm repurposes the covariance matrix from the original CMA-ES algorithm into a guide for the population into more favorable regions within the search space and an indicator of stagnation when the matrix stops changing or evolving through several iterations. To detect stagnation, the following equation is evaluated independently for each sub-population, each iteration:

$$\Delta\mathbf{C} = |\mathbf{C}_\tau - \mathbf{C}_{\tau+1}| \tag{8}$$

The $\Delta\mathbf{C}$ variable represents the amount of change between one iteration and the next. Its value is calculated for each distinct sub-population through matrix subtraction between the current and the following covariance matrices. To differentiate between significant and insignificant change, a threshold of 0.003 is considered. Any variation between the matrices with a value under the threshold is considered insignificant. If, as a result of the subtraction, significant changes were found in only 10 elements or less of the whole covariance matrix, that specific sub-population is considered stagnated.

For the removal, the group detected as stagnated is eliminated from the population and never considered again for future iterations. Before removing the stagnated sub-population, one verification ensures that the sub-population with the best current fitness is always preserved.

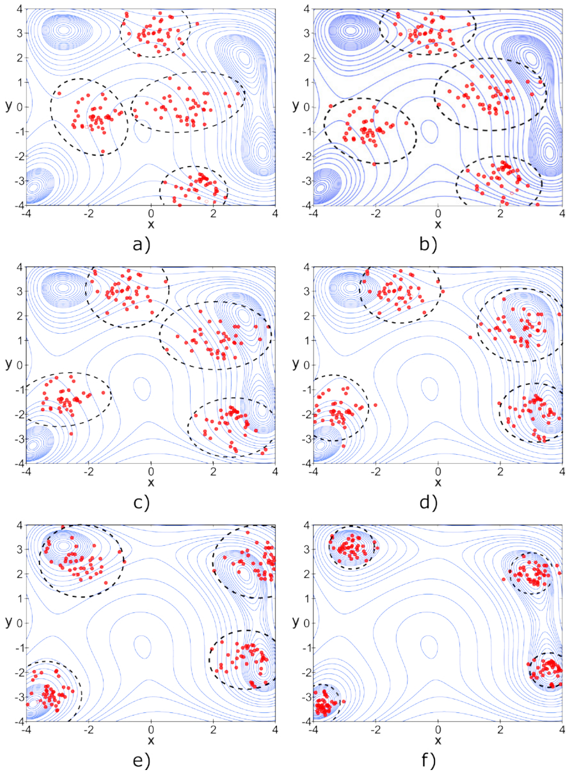


Fig. 2. Illustration of the multiple sub-populations optimizing the search space

4. Experimental results

4.1. Friedman test

Recently, the use of statistical analysis as a process for evaluating an improvement of a metaheuristic algorithm has become a widely used technique in the field of computational intelligence [4]. Since statistical analysis is necessary for a good comparison of metaheuristic techniques due to their stochastic nature [16], a non-parametric statistical method is used in this paper. The Friedman test is a non-parametric statistical method that uses ranked data [5]. This non-parametric statistical test has been conducted to demonstrate the effectiveness of the MCMAES-SC proposal. The results of this Friedman test can be seen in the table 1, where it can be seen in the ranking as the proposal achieves the best ranking. Also, in the table 1, the p-value essentially proves that there are significant differences between the algorithms examined, with a value of $2.89E-37$.

Table 1. Friedman Rank Test 150 Dims

Algorithm	Friedman Rank	Final Ranking
MCMAES-SC	2.04E+00	1
CMAES	2.50E+00	2
AGPSO	7.54E+00	8
BLPSO	4.26E+00	4
DE	8.58E+00	9
GA	5.33E+00	5
JADE	5.38E+00	6
LSHADE	3.29E+00	3
PSO	6.07E+00	7
p-value	2.89E-37	

An Achilles' heel of metaheuristic algorithms is their susceptibility to low performance when optimizing high-dimensional problems, so their scalability is compromised [7]. For this reason, to test the feasibility and applicability of the proposed MCMAES-SC algorithm, experiments were performed on a set of functions of different classes in 150 dimensions. The four types of functions proposed for the experiments test different characteristics of the designed algorithms, for example the multi-modal type (Table 2) tests the ability of the metaheuristic algorithms to maintain a healthy diversity in their solutions to escape from local minima and to obtain a good fitness, the unimodal functions (Table 3) are relatively simple functions and the purpose of these is to test the ability of the algorithms to refine the solutions as much as possible allowing to evaluate the algorithm in terms of accuracy, the composite functions (Table 4) are functions that combine unimodal and multi-modal functions in order to test the algorithm in larger search spaces and with greater difficulty since they will have to search among several local minima and at the same time refine the solutions as much as possible, finally, the algorithm is also evaluated in functions of shifted type (Table 5), these functions are considered because there is an error in the design of metaheuristic algorithms called "zero bias," which compromises the performance of the algorithm by biasing it to numerical domains where the positions of the search agents contain the value 0 as part of their optimal solution, for this reason, to check that the algorithm does not have the mentioned error, the translated functions move the positions of their global optima to avoid having a value 0 as part of the solution vector. The purpose of the Tables (2-5) is to show two measures of central tendency: the average best fitness of the performed runs (AB), highlighted in bold, and the standard deviation associated with the same data (SD), highlighted in blue. In Table 2 of multi-modal functions, it is evident that MCMAES-SChas most of the best average values in the 20 test functions proposed for this experiment, obtaining 16/20 best values in terms of AB, tied in 3 functions and losing in 4 to its predecessor CMAES. On the other hand, in terms of standard deviation highlighted in blue, MCMAES-SCshows a very consistent performance, observing very low standard deviation (SD) values, which demonstrates that its variability is low from one run to another.

Table 3 shows the results in 5 uni-modal functions; as previously mentioned, these functions test the capacity of the algorithms to exploit and refine their solutions in the search space, taking into account that the functions increase their complexity by having 150 dimensions we can observe that MCMAES-SC has an excellent performance obtaining 3/5 best values in terms of AB, no tie, and losing in 2 of the functions proposed with potent algorithms such as JADE and LSHADE. Table 4 shows the results in the class of composite functions; for this experiment, we considered 4 functions that test the robustness of the algorithms since they combine two or more classes of functions, making them challenging to optimize; in the results, we can observe that MCMAES-SC obtains 2/4 of the best values in terms of AB and SD, and losing in 2 against algorithms such as GA and LSHADE. Finally, Table 5 shows 7 translated functions, which, as mentioned before, shows that the proposed algorithm MCMAES-SC does not have the zero bias error in the table 6/7 of the best values in terms of AB and SD are obtained, which shows that MCMAES-SC does not have this design error.

It is always important to show the convergence plots to check the status of the optimization process; for this reason, we decided to place 2 representative samples of each class of functions in Figure 3. In Figure 3, subfigures (a) and (b) are two samples of multi-modal functions where we can observe the significant improvement of the original CMAES method versus the multi-population algorithm MCMAES-SC, which shows that it favors the scalability of the algorithm in high dimensions offering better optimization times and better solutions taking into consideration the fitness and the function accesses used, subfigures (c) and (d) show two uni-modal functions showing the result of the optimization process in this class of functions, showing excellent quality in the exploitation process compared to the rest of the competing algorithms. In subfigures (e) and (f), two samples of graphs in composite functions are shown, and what can be observed is the low performance of several popular algorithms that are limited by the large number of dimensions to be optimized. At the same time, the MCMAES-SC proposal keeps improving its solutions throughout the optimization process. Finally, subfigures (g) and (h) show all the algorithms in two samples of shifted functions where the main objective is to show that a particular algorithm is not affected in functions that do not have their optimal positions in a value of 0, so it is evident for the results on the graph that the objective is met because if we look in detail when comparing the graphs of the shifted and unshifted functions are similar, which shows that the algorithm and the competitors do not have the zero bias design error.

Table 2. Statistical results for benchmark multi-modal functions in 150 dimensions

Function		MCMAES-SC	CMAES	AGPSO	BLPSO	DE	GA	JADE	LSHADE	PSO
F1.Ackley	AB	**1.40E-14**	1.31E-07	1.26E+01	4.03E-04	1.33E+01	4.23E-02	6.51E+00	7.32E-05	2.24E-01
	SD	**9.01E-16**	2.15E-08	1.33E+00	3.31E-04	6.26E-01	6.13E-03	3.76E+00	1.90E-05	3.49E-01
F2.Dixon	AB	**6.67E-01**	6.67E-01	7.80E+03	2.24E+01	1.69E+06	7.69E+01	2.38E+01	6.67E-01	5.76E+01
	SD	**0.00E+00**	4.30E-04	3.11E+04	2.06E+01	6.62E+05	1.68E+01	1.55E+01	3.42E-04	1.91E+01
F3.Griewank	AB	**0.00E+00**	3.87E-09	6.28E+00	2.88E-03	3.61E+02	8.46E-02	2.02E-02	2.47E-04	3.77E-02
	SD	**0.00E+00**	1.31E-09	2.29E+01	8.34E-03	7.39E+01	4.02E-02	5.13E-02	1.35E-03	4.77E-02
F4.Infinity	AB	**6.22E-105**	1.98E-41	3.22E-12	1.07E-14	3.12E-01	2.27E-15	3.42E-16	1.76E-26	6.06E-11
	SD	**1.26E-104**	2.39E-41	7.27E-12	1.41E-14	1.29E-01	4.24E-15	1.83E-15	2.61E-26	3.98E-11
F5.Levy	AB	2.09E-02	**4.64E-14**	8.78E+01	1.96E-01	8.18E+01	8.18E-04	1.61E+01	1.74E-08	7.27E+00
	SD	3.85E-02	**1.70E-14**	1.66E+01	2.04E-01	2.10E+01	2.39E-04	8.26E+00	1.11E-08	3.71E+00
F6.Mishra11	AB	**0.00E+00**	3.29E-10	3.86E-04	1.42E-10	5.60E-03	1.62E-12	1.58E-30	2.21E-12	5.18E-13
	SD	**0.00E+00**	2.45E-10	4.60E-04	3.52E-10	2.91E-03	1.54E-12	1.60E-30	3.24E-12	1.15E-12
F7.MultiModal	AB	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	4.44E-308	**0.00E+00**	7.15E-178
	SD	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
F8.Plateau	AB	**3.00E+01**	**3.00E+01**	4.88E+01	**3.00E+01**	8.94E+01	**3.00E+01**	3.15E+01	**3.00E+01**	3.71E+01
	SD	**0.00E+00**	**0.00E+00**	6.13E+00	**0.00E+00**	1.13E+01	**0.00E+00**	1.87E+00	**0.00E+00**	2.57E+00
F9.Powell	AB	**7.23E-05**	2.70E-03	6.04E+02	5.48E-01	4.69E+03	8.36E-01	2.78E-02	9.30E-03	3.40E-01
	SD	**1.42E-05**	5.05E-04	9.62E+02	1.95E-01	1.32E+03	1.31E-01	2.08E-02	2.86E-03	8.54E-02
F10.Qing	AB	**9.09E-26**	6.35E-02	1.14E+04	6.87E+00	4.17E+10	3.50E+02	3.64E+01	5.14E+00	9.65E+02
	SD	**5.62E-26**	1.95E-01	9.71E+03	1.40E+01	1.27E+10	1.50E+02	1.23E+02	3.57E+00	1.11E+03
F11.Quartic	AB	6.23E+01	**6.03E+01**	1.23E+02	6.20E+01	1.83E+02	7.86E+01	8.81E+01	6.06E+01	6.81E+01
	SD	**7.80E-01**	1.11E+00	4.86E+01	9.35E-01	4.61E+01	2.47E+00	5.21E+00	1.03E+00	1.63E+00
F12.Quintic	AB	5.66E-02	**1.45E-04**	6.15E+01	8.25E-02	2.91E+04	1.33E+00	6.15E-01	2.12E+00	4.05E-01
	SD	3.10E-01	**4.80E-05**	1.60E+01	9.91E-02	1.08E+04	1.88E-01	7.66E-01	1.50E+00	2.36E-01
F13.Rosenbrock	AB	**1.24E+02**	1.43E+02	8.81E+04	3.78E+02	4.19E+05	2.44E+02	2.96E+02	1.42E+02	3.22E+02
	SD	2.80E+01	2.01E+01	8.75E+04	5.08E+01	1.41E+05	7.75E+01	8.41E+01	**7.46E-01**	8.36E+01
F14.Schwefel21	AB	**2.38E-08**	2.42E-03	5.76E+01	2.74E+01	5.98E+01	2.45E+01	8.30E+01	3.69E-01	9.97E+00
	SD	**6.90E-09**	5.20E-04	2.87E+00	2.94E+00	4.10E+00	3.30E+00	2.85E+00	6.45E-02	7.40E-01
F15.Schwefel22	AB	**6.72E-14**	1.61E-04	4.40E+03	4.93E-02	1.67E+03	1.45E+00	2.35E+01	1.36E-01	4.22E+58
	SD	**1.40E-13**	3.06E-05	3.56E+02	8.61E-02	2.93E+02	2.17E-01	7.49E+00	5.09E-02	2.31E+59
F16.Schwefel26	AB	**-3.08E+05**	-2.34E+05	-4.11E+04	-5.94E+04	-4.34E+04	-5.43E+04	-4.07E+04	-3.75E+04	-3.07E+04
	SD	5.68E+04	6.25E+04	1.71E+03	7.33E+02	1.49E+03	7.38E+02	8.10E+03	**6.20E+02**	3.20E+03
F17.Step	AB	**0.00E+00**	**0.00E+00**	1.37E+02	6.50E+00	4.17E+04	**0.00E+00**	9.83E+01	**0.00E+00**	1.77E+01
	SD	**0.00E+00**	**0.00E+00**	4.13E+01	4.28E+00	7.79E+03	**0.00E+00**	1.08E+02	**0.00E+00**	6.26E+00
F18.Stybtang	AB	-5.28E+03	-5.67E+03	-4.93E+03	-5.78E+03	-4.73E+03	-4.93E+03	-5.31E+03	**-5.85E+03**	-4.78E+03
	SD	6.42E+01	4.62E+01	9.05E+01	2.95E+01	1.03E+02	9.16E+01	3.18E+02	**1.96E+01**	5.31E+02
F19.Trid	AB	**-5.58E+05**	-1.58E+05	8.41E+07	5.02E+06	1.35E+09	1.42E+07	1.29E+06	3.65E+04	6.04E+06
	SD	**1.98E+04**	2.95E+05	1.01E+08	1.55E+06	2.42E+08	5.50E+06	8.18E+05	4.09E+04	2.09E+06
F20.Vincent	AB	**-4.20E+14**	-2.12E+14	-1.44E+02	-1.50E+02	-1.42E+02	-1.50E+02	-1.49E+02	-1.21E+02	-1.49E+02
	SD	9.98E+13	6.34E+13	3.41E+00	1.22E-01	1.57E+00	**5.87E-04**	1.11E-01	1.62E+00	1.51E+00

Table 3. Statistical results for benchmark unimodal functions in 150 dimensions

Function		MCMAES-SC	CMAES	AGPSO	BLPSO	DE	GA	JADE	LSHADE	PSO
F21.Rothyp	AB	**5.79E-30**	4.23E-07	6.96E+04	4.55E-03	1.03E+06	4.85E+00	4.67E-03	2.07E-04	9.53E-01
	SD	**8.10E-30**	1.72E-07	1.47E+05	1.26E-02	2.19E+05	1.64E+00	1.38E-02	1.23E-04	6.89E-01
F22.Schwefel2	AB	9.74E+02	2.49E+04	2.89E+07	6.40E-01	2.04E+08	1.23E+03	**1.87E-02**	1.47E+01	9.52E+02
	SD	1.96E+03	1.78E+04	3.61E+07	2.93E+00	4.08E+07	6.05E+02	**6.46E-02**	2.22E+01	4.09E+03
F23.Sphere	AB	**1.56E-42**	3.08E-15	3.50E+00	2.54E-07	1.04E+02	4.08E-04	1.96E-08	1.09E-09	1.04E-04
	SD	**1.52E-42**	1.34E-15	9.06E+00	8.89E-07	2.09E+01	1.13E-04	8.56E-08	6.20E-10	5.23E-05
F24.Sum2	AB	**1.00E-32**	1.91E-09	1.58E+03	1.37E-05	2.47E+04	1.10E-01	7.47E-06	4.76E-06	3.16E-02
	SD	**1.62E-32**	1.00E-09	2.20E+03	2.09E-05	5.22E+03	4.03E-02	2.42E-05	3.17E-06	4.25E-02
F25.Sumpow	AB	2.04E-09	6.28E-10	8.96E-27	1.71E-38	6.99E-04	4.71E-09	5.29E-07	**5.24E-51**	2.77E-44
	SD	9.83E-10	3.24E-10	2.51E-26	7.43E-38	1.91E-03	1.10E-08	1.14E-06	**5.75E-51**	1.39E-43

5. Conclusions

This paper proposes a new strategy to improve the management of the population in the search space. The method inspires the covariance matrix mechanism of the CMA-ES algorithm. The proposed MCMAES-SC algorithm adapts the fitness landscape using multiple covariance matrices for different sub-populations. These are analyzed iteratively to identify and eliminate stagnated subgroups, freeing up function evaluations for the remaining population. To evaluate

Table 4. Statistical results for benchmark composite functions in 150 dimensions

Function		MCMAES-SC	CMAES	AGPSO	BLPSO	DE	GA	JADE	LSHADE	PSO
F26.Hybrid1	AB	**6.16E-19**	3.83E-06	4.12E+04	2.22E-03	4.56E+04	4.15E-01	5.61E-02	2.32E-03	1.56E-01
	SD	**3.54E-19**	7.37E-07	2.30E+04	2.78E-03	1.21E+04	9.37E-02	1.24E-01	1.04E-03	1.20E-01
F27.Hybrid2	AB	2.99E+02	1.51E+02	1.02E+03	1.77E+02	1.89E+03	**1.49E+02**	4.63E+02	1.50E+02	6.73E+02
	SD	3.37E+01	4.31E+00	1.31E+02	1.27E+01	2.50E+02	**1.07E-01**	2.76E+02	2.49E+00	6.42E+01
F28.Hybrid3	AB	**1.65E+02**	1.72E+02	1.90E+03	1.95E+03	1.38E+08	5.81E+03	3.85E+03	1.86E+02	3.30E+03
	SD	**8.17E-01**	2.68E+00	2.15E+02	2.49E+02	5.79E+07	5.61E+02	8.73E+02	3.47E+00	4.55E+02
F29.Hybrid4	AB	2.82E+02	1.50E+02	1.63E+03	1.74E+02	1.89E+03	1.50E+02	2.02E+02	**1.49E+02**	7.35E+02
	SD	3.45E+01	3.11E+00	5.32E+02	1.66E+01	2.96E+02	3.35E-01	6.58E+01	**9.95E-04**	6.84E+01

Table 5. Statistical results for benchmark shifted functions in 150 dimensions

Function		MCMAES-SC	CMAES	AGPSO	BLPSO	DE	GA	JADE	LSHADE	PSO
F30.ShiftedAckley	AB	**1.47E-14**	1.37E-07	1.26E+01	3.41E-04	1.33E+01	4.09E-02	6.34E+00	6.83E-05	3.65E-01
	SD	**1.23E-15**	3.03E-08	1.28E+00	3.37E-04	9.25E-01	6.83E-03	3.12E+00	1.49E-05	3.83E-01
F31.ShiftedPowell	AB	**7.15E-05**	2.88E-03	7.21E+02	6.26E-01	4.52E+03	8.35E-01	2.89E-02	9.05E-03	3.56E-01
	SD	**1.21E-05**	4.81E-04	1.29E+03	2.56E-01	1.46E+03	1.26E-01	2.12E-02	2.31E-03	7.50E-02
F32.ShiftedRosenbrock	AB	1.22E+02	1.41E+02	1.73E+05	2.82E+02	8.31E+05	**1.11E+02**	2.91E+02	1.45E+02	1.04E+03
	SD	1.77E+01	1.66E+01	6.55E+04	8.21E+01	2.48E+05	5.95E+01	8.62E+01	**1.09E+01**	4.00E+03
F33.ShiftedRothyp	AB	**8.72E-26**	4.93E-07	5.06E+04	1.22E-03	1.09E+06	4.15E+00	3.31E-04	2.43E-04	2.77E+00
	SD	**6.57E-27**	2.70E-07	8.79E+04	3.79E-03	2.30E+05	1.23E+00	1.47E-03	1.08E-04	8.13E+00
F34.ShiftedSchwefel22	AB	**3.85E-13**	1.51E-04	2.00E+03	8.29E-02	1.63E+03	1.54E+00	2.21E+01	1.53E-01	2.50E+02
	SD	**2.48E-13**	4.13E-05	5.48E+02	1.07E-01	3.63E+02	2.67E-01	6.94E+00	5.89E-02	5.94E+02
F35.ShiftedSphere	AB	**1.08E-27**	3.67E-15	8.76E-01	3.13E-06	1.04E+02	4.12E-04	1.86E-07	1.04E-09	9.14E-05
	SD	**7.09E-29**	1.19E-15	4.79E+00	1.69E-05	2.84E+01	1.56E-04	7.07E-07	4.44E-10	6.21E-05
F36.ShiftedSum2	AB	**8.05E-26**	1.80E-09	1.40E+03	1.81E-05	2.57E+04	1.25E-01	1.98E-05	4.32E-06	3.61E-02
	SD	**5.32E-27**	8.02E-10	2.13E+03	2.91E-05	5.65E+03	7.70E-02	9.20E-05	2.90E-06	4.27E-02

the feasibility and applicability of the proposed MCMAES-SC algorithm, experiments were conducted on a range of functions belonging to different classes in 150 dimensions. The experimental set comprises 36 well-known and frequently used test functions, comprising uni-modal, multi-modal, composite, and shifted functions.

Moreover, the performance of the proposed algorithm is statistically evaluated and compared with CMA-ES, DE, JADE, LSHADE, PSO, AGPSO, BLPSO, and GA algorithms. The statistical results can conclude that the MCMAES-SC algorithm positively enhances the standard version of CMA-ES and is also competitive with other sophisticated evolutionary algorithms such as LSHADE and JADE. In turn, the visual results of the convergence plots indicate that the performance of the MCMAES-SC improves the local minima stagnation problem and makes better use of function accesses, constantly improving its solutions and providing better optimization time. Future work aims to explore potential areas for research to refine the existing method for improving the management of the population's function accesses. Possible improvements to other metaheuristic algorithms, such as DE, GA, or PSO, will also be tested. Furthermore, it would be pertinent to expand the scope of the experiments to include real-world engineering problems.

References

[1] Auger, A., Hansen, N., 2005. A restart cma evolution strategy with increasing population size, in: 2005 IEEE congress on evolutionary computation, IEEE. pp. 1769–1776.

[2] Bäck, T., Foussette, C., Krause, P., 2013. Contemporary evolution strategies. volume 86. Springer.

[3] Bäck, T., Schwefel, H.P., 1993. An overview of evolutionary algorithms for parameter optimization. Evolutionary computation 1, 1–23.

[4] Derrac, J., García, S., Molina, D., Herrera, F., 2011. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation 1, 3–18.

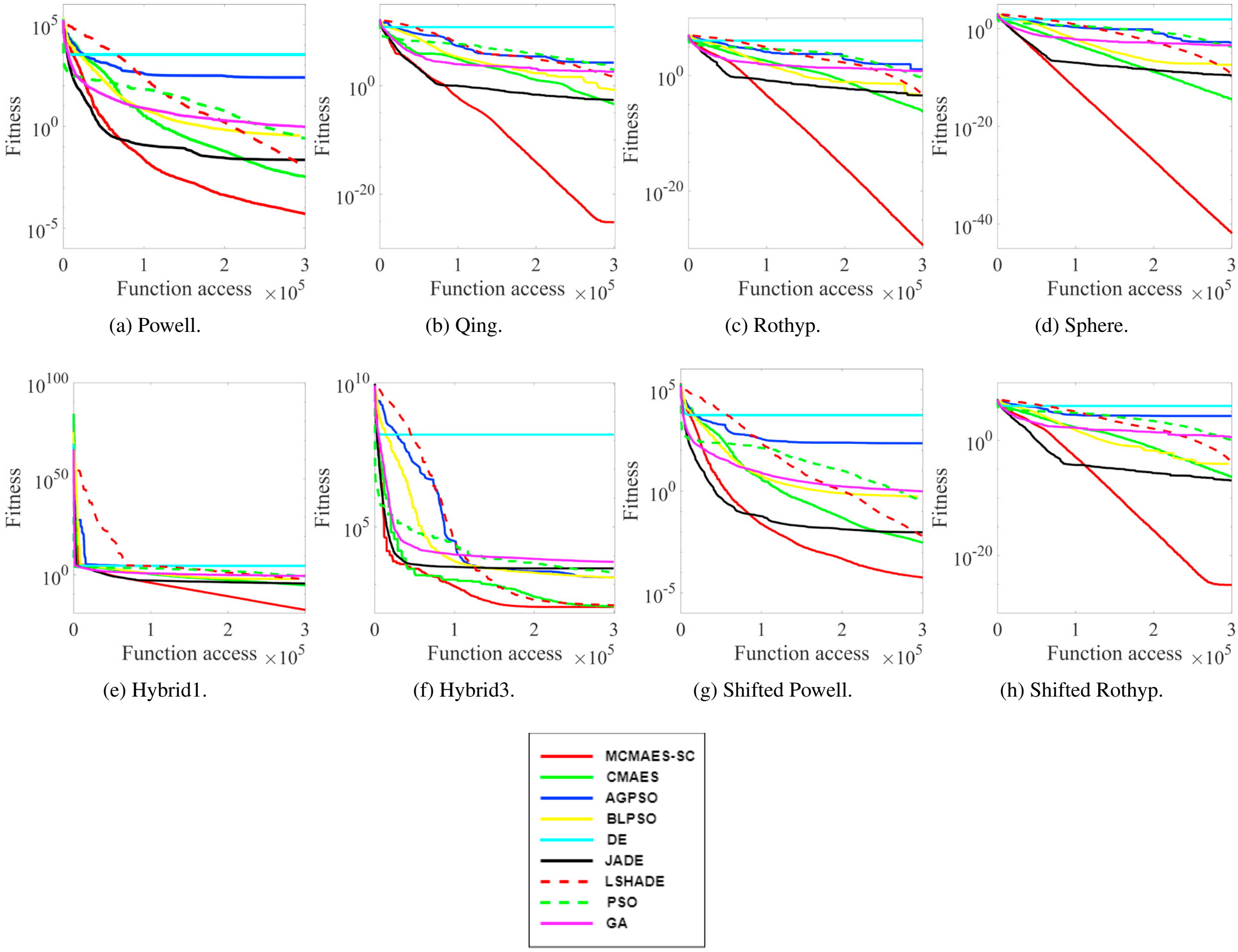
[5] Friedman, M., 1937. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the american statistical association 32, 675–701.

[6] Friedrichs, F., Igel, C., 2005. Evolutionary tuning of multiple svm parameters. Neurocomputing 64, 107–117.

[7] Halim, A.H., Ismail, I., Das, S., 2021. Performance assessment of the metaheuristic optimization algorithms: an exhaustive review. Artificial Intelligence Review 54, 2323–2409.

[8] Hansen, N., 2006. The cma evolution strategy: a comparing review. Towards a new evolutionary computation: Advances in the estimation of distribution algorithms , 75–102.

[9] Hansen, N., 2016. The cma evolution strategy: A tutorial. arXiv preprint arXiv:1604.00772 .



(a) Powell. (b) Qing. (c) Rothyp. (d) Sphere.

(e) Hybrid1. (f) Hybrid3. (g) Shifted Powell. (h) Shifted Rothyp.

(i) Color representation for each algorithm.

Fig. 3. A comparative analysis of the converge curves.

[10] Hansen, N., Ostermeier, A., 2001. Completely derandomized self-adaptation in evolution strategies. Evolutionary computation 9, 159–195.

[11] Jin, J., Yang, C., Zhang, Y., 2020. An improved cma-es for solving large scale optimization problem, in: Advances in Swarm Intelligence: 11th International Conference, ICSI 2020, Belgrade, Serbia, July 14–20, 2020, Proceedings 11, Springer. pp. 386–396.

[12] Kämpf, J.H., Robinson, D., 2009. A hybrid cma-es and hde optimisation algorithm with application to solar energy potential. Applied Soft Computing 9, 738–745.

[13] Liang, Y., Wang, X., Zhao, H., Han, T., Wei, Z., Li, Y., 2019. A covariance matrix adaptation evolution strategy variant and its engineering application. Applied Soft Computing 83, 105680.

[14] Maki, A., Sakamoto, N., Akimoto, Y., Nishikawa, H., Umeda, N., 2020. Application of optimal control theory based on the evolution strategy (cma-es) to automatic berthing. Journal of Marine Science and Technology 25, 221–233.

[15] Moosavian, H., Mesbahi, P., Moosavian, N., Daliri, H., 2021. Optimal design of truss structures with frequency constraints: a comparative study of de, ide, lshade, and cmaes algorithms. Engineering with Computers , 1–19.

[16] Mousavirad, S.J., Ebrahimpour-Komleh, H., 2017. Human mental search: a new population-based metaheuristic optimization algorithm. Applied Intelligence 47, 850–887.

[17] Navarro, M.A., Oliva, D., Ramos-Michel, A., Morales-Castañeda, B., Zaldívar, D., Luque-Chang, A., 2022. A review of the use of quasi-random number generators to initialize the population in meta-heuristic algorithms. Archives of Computational Methods in Engineering 29, 5149–5184.

[18] Wu, Y., Peng, X., Wang, H., Jin, Y., Xu, D., 2022. Cooperative coevolutionary cma-es with landscape-aware grouping in noisy environments. IEEE transactions on evolutionary computation .