ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

Enhancing Evolution Strategies with Evolution Path Bias

Oliver Kramer

Computational Intelligence Lab
Department of Computer Science
Carl-von-Ossietzky University of Oldenburg
26111 Oldenburg, Germany
oliver.kramer@uni-oldenburg.de

Abstract. Evolution Strategies (ES) have emerged as a powerful and effective method for optimization and reinforcement learning tasks, largely due to their simplicity and scalability. However, current ES techniques can be limited in their capacity to quickly converge on the optimal solution. In this paper, we propose a novel approach to enhance ES by incorporating an evolution path-informed bias in the Gaussian mutation operator. This bias is designed to facilitate faster descent on decreasing functions. Our method leverages the evolution path, which represents the historical search directions, to intelligently bias the Gaussian mutation. By doing so, it enables the algorithm to be more sensitive to the underlying function's structure and adaptively exploit this information for more efficient exploration. We validate our approach through experiments on three benchmark functions: a linear function, we call Downhill function here, a Parabolic ridge, and a Sphere function. The results demonstrate that our evolution path-informed bias significantly accelerates convergence on in most of the cases.

1 Introduction

Evolution Strategies (ES) have emerged as powerful optimization techniques for tackling continuous optimization problems due to their robustness, simplicity, and adaptability. These strategies are based on the principles of natural selection and genetic variation, and have been successfully applied to a wide range of complex real-world problems. One of the three main design principles of ES suggests that mutation operators should not exhibit any bias to ensure an unbiased exploration of the search space. However, studies have shown that introducing bias in mutation operators can be advantageous in certain situations and lead to improved convergence and exploration capabilities[5].

In this paper, we propose an enhanced $(1, \lambda)$-ES that combines the use of evolution paths and biased mutation operators to improve the performance of the algorithm on a set of ridge functions. Ridge functions are a class of optimization problems that exhibit a narrow and elongated global optimum region, which pose a significant challenge for optimization algorithms. The approach aims to harness the benefits of biased mutation operators while exploiting the guidance provided by the evolution path to effectively explore and converge to the global optimum.

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

The remainder of the paper is organized as follows: Section 2 will discuss the role of biased mutation operators in evolution strategies. Section 3 will focus on the biased mutation operator based on evolution paths. Section 4 will present an experimental analysis. Finally, Section 5 will conclude the paper with a summary of the findings and suggestions for future research directions.

2 ES and Biased Mutation

ES can be characterized by their population size and selection mechanisms. One of the widely used variants of ES is the $(1, \lambda)$-ES, which has a parent population size of one and an offspring population size of λ. The algorithm iteratively produces offspring by mutating the parent solution and selecting the best individual from the offspring population to replace the parent in the next generation.

The $(1, \lambda)$-ES incorporates a mutation rate control mechanism to adaptively adjust the step size during the optimization process. Rechenberg's 1/5th success rule is a popular approach for mutation rate control in $(1, \lambda)$-ES. According to this rule, the step size is increased if the success rate of generating better offspring is higher than 1/5th and decreased otherwise. This adaptive control mechanism allows the algorithm to balance the exploration and exploitation trade-off effectively. In our biased mutation ES we implement the rule by increasing the step size by $\exp(1/\lambda)$ in case of success and decreasing it by $\exp(-1/5)$ otherwise notated as:

$$\sigma := \sigma \cdot \exp([1/\lambda] \vee [-1/5]) \tag{1}$$

Despite the general design principle in ES suggesting that mutation operators should not introduce any bias, research has shown that biased mutation operators can be beneficial in specific situations. Biased mutation operators can improve the exploration and convergence properties of an optimization algorithm by guiding the search process towards promising regions of the search space. In this section, we discuss different types of biased mutation operators and their potential advantages.

A Gaussian biased mutation operator introduces a bias in the mutation by sampling from a Gaussian distribution with a non-zero mean. The non-zero mean represents the direction of the bias, and the standard deviation determines the magnitude of the bias. This type of biased mutation operator can effectively guide the search process when the problem exhibits a known gradient or structure that can be exploited. Biased Gaussian mutation has been implemented in a self-adaptive variant for unconstrained [4] and contrained problems [5].

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9.
Available from http://www.i6doc.com/en/.

3 Incorporating Evolution Path Bias in Gaussian Mutation

3.1 Evolution Path

Non-local information about the population can improve optimization performance by guiding the search process towards promising regions in the search space. An evolution path, denoted by $\mathbf{s}$, is a generational record of step sizes employed during the optimization process, which serves as a source of non-local information [6, 3, 1]. The evolution path, also known as the cumulative path length, can provide insights into the correlation of search directions, thereby enabling more informed mutation steps.

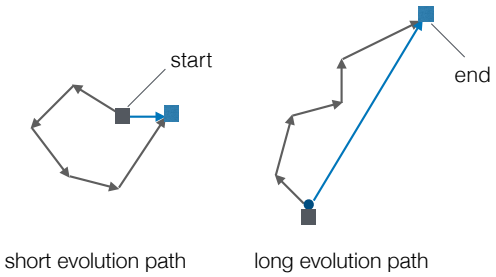

Fig. 1: Principle of evolution path in space of $\mathbf{z}$-mutations.

In the context of the $(1, \lambda)$-ES with Gaussian mutation, the evolution path $\mathbf{s}$ is updated using the successful mutation vector $\mathbf{z}_k$. The update rule for the evolution path is:

$$\mathbf{s} := (1 - c) \cdot \mathbf{s} + c \cdot \mathbf{z}_k \tag{2}$$

with c balancing between the past directions and the current. This update rule ensures that information from previous steps is not lost, and the correlation of directions $\mathbf{z}_k$ determines the length of the path $\mathbf{s}$. Single steps pointing in similar directions will increase the path, while opposing directions will shorten the path, as illustrated in Figure 1. In CMA-ES [2] and cumulative path length control ES [6, 3, 1], the path length $|\mathbf{s}|$ is used to determine a global step size σ.

3.2 Biased Gaussian Mutation with Evolution Path

In this paper, we propose to incorporate the evolution path as a bias in Gaussian mutation. The biased Gaussian mutation equation is given by:

$$\mathbf{x}_k = \mathbf{x} + \sigma \cdot (\gamma \cdot \mathbf{z}_k + (1 - \gamma) \cdot \mathbf{s}) \tag{3}$$

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

Here, $\mathbf{x}_k$ is a mutated solution, $\mathbf{x}$ is the original solution, σ is the step size, $\mathbf{z}_k$ is the Gaussian random vector sampled from $\mathcal{N}(\mathbf{0}, \mathbf{1})$, and $\mathbf{s}$ is the evolution path. Parameter γ weighs the generated random vector and the bias.

By incorporating the evolution path as a bias in the Gaussian mutation, our proposed method aims to exploit the non-local information and correlations in the search directions to improve the convergence and exploration properties of the $(1, \lambda)$-ES.

Algorithm 1 provides a pseudocode implementation of the proposed method, which combines the evolution path update and biased Gaussian mutation for enhancing the performance of the $(1, \lambda)$-ES. Step size σ is controlled using a per-generation update variant of the Rechenberg 1/5 rule defining success.

Algorithm 1: Biased $(1, \lambda)$-ES

given $c \approx \sqrt{1/(N+1)}$
initialize $\mathbf{s} = \mathbf{0}, \sigma \in \mathbb{R}_+^N, \mathbf{x} \in \mathbb{R}^N$
repeat
  for $k \in \{1, \ldots, \lambda\}$ **do**
    $\mathbf{z}_k = \mathcal{N}(\mathbf{0}, \mathbf{1})$
    $\mathbf{x}_k = \mathbf{x} + \sigma \cdot (\gamma \cdot \mathbf{z}_k + (1 - \gamma) \cdot \mathbf{s})$
  end for
  $(\mathbf{x}, \mathbf{z}) \leftarrow$ select best of $\{(\mathbf{x}_k, \mathbf{z}_k) | 1 \leq k \leq \lambda\}$
  $\sigma := \sigma \cdot \exp([1/\lambda] \vee [-1/5])$
  $\mathbf{s} := (1 - c) \cdot \mathbf{s} + c \cdot \mathbf{z}$
until termination condition

4 Experiments

We conducted experiments for each benchmark function in 20 and 100 dimensions, with 100 repetitions to assess the robustness and consistency of the proposed ES, see Appendix A. The results were compared with a traditional $(1, \lambda)$-ES without biased mutation. The parameters for both algorithms were initialized as follows:

Population size $\lambda = 10 \cdot N$, initial step size $\sigma = 1.0$, per generation mutation rate control with $\tau = 1/\lambda$, and the biased Gaussian mutation operator.

To evaluate the performance of the algorithms, we used the best objective function value found during the evolutionary run after 1000 fitness function evaluations. The experimental results were analyzed using the Wilcoxon rank-sum test to assess the statistical significance of the observed differences in performance between the proposed ES with biased mutation operators and the traditional $(1, \lambda)$-ES.

The results indicate that the proposed ES with biased mutation operators outperforms ($\uparrow$) the traditional $(1, \lambda)$-ES in four of six cases in terms of mean fitness. On the Downhill function with $N = 20$ the advantage was not signif-

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

	Non-biased $(1, \lambda)$-ES			
Function	20D		100D	
Downhill	-3.6e5 ± 2.6e3		-5.5e5 ± 179.1	
Parabolic	-356.1 ± 9.40		-86.8 ± 3.12	
Sphere	9.8e-51 ± 2.6e-50		5.2e-20 ± 5.6e-20	
	Biased $(1, \lambda)$-ES			
Function	20D		100D	
Downhill	-3.6e5 ± 2.2e3	✗	-5.5e5 ± 194.8	↑ ✓
Parabolic	-649.1 ± 19.31	↑ ✓	-132.6 ± 4.96	↑ ✓
Sphere	1.9e-56 ± 1.2e-55	↑ ✓	2.2e-17 ± 3.1e-17	↓ ✓

Table 1: Experimental results comparing the biased $(1, \lambda)$-ES and the non-biased $(1, \lambda)$-ES with $\gamma = 0.5$. The arrows indicate improvement ↑ or deterioration ↓ , while ✓indicates statistical significance.

icant, while on the Sphere with $N = 100$ the unbiased $(1, \lambda)$-ES was superior. The Wilcoxon test confirms that the observed improvements are statistically significant in five cases, marked with ✓.

These findings demonstrate the effectiveness of incorporating biased mutation operators with evolution paths in enhancing the performance of ES algorithms on challenging optimization problems, such as ridge functions and the Sphere function.

5 Conclusions

In this paper, we investigated the incorporation of biased mutation operators, specifically using the evolution path, to enhance the performance of ES for continuous optimization problems. We proposed a method that combines the $(1, \lambda)$-ES with biased Gaussian mutation using the evolution path as a source of non-local information.

Our experimental results, obtained by testing the proposed method on three benchmark functions across different dimensions, demonstrated that the evolution path-based bias in Gaussian mutation outperforms the traditional $(1, \lambda)$-ES on the Downhill and ridge function, while causing no harm on the Sphere function. The improvements in terms of best fitness, median fitness, and convergence rate were statistically significant, as confirmed by the Wilcoxon rank-sum test.

As future work, we suggest exploring other biased mutation operators and investigating their potential advantages in different problem domains. Furthermore, the proposed method could be extended to other variants of ES or combined with other optimization techniques to create hybrid algorithms with improved convergence and exploration properties.

ESANN 2023 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 4-6 October 2023, i6doc.com publ., ISBN 978-2-87587-088-9. Available from http://www.i6doc.com/en/.

A Benchmark Functions

The selected benchmark functions are defined as follows:
Downhill function:

$$f_1(x) = -\sum_{i=1}^{N} x_i \tag{4}$$

Parabolic ridge:

$$f_3(x) = x_1^2 + \sum_{i=2}^{N} (x_i - x_1)^2 \tag{5}$$

Sphere function:

$$f_4(x) = \sum_{i=1}^{N} x_i^2 \tag{6}$$

In these functions, $\mathbf{x} = (x_1, x_2, ..., x_N)$ is the decision variable vector, N is the number of dimensions.

References

[1] H.-G. Beyer and D. V. Arnold. Qualms regarding the optimality of cumulative path length control in CSA/CMA-evolution strategies. *Evolutionary Computation*, 11(1):19–28, 2003.

[2] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. In *IEEE Congress on Evolutionary Computation, (CEC)*, pages p. 1–8. IEEE, 2001.

[3] N. Hansen, A. Ostermeier, and A. Gawelczyk. On the adaptation of arbitrary normal mutation distributions in evolution strategies: The generating set adaptation. In L. J. Eshelman, editor, *International Conference on Genetic Algorithms (ICGA)*, pages 57–64. Morgan Kaufmann, 1995.

[4] L. Hildebrand, B. Reusch, and M. Fathi. Directed mutation-a new self-adaptation for evolution strategies. In *IEEE Congress on Evolutionary Computation, (CEC)*, pages 1550–1557. IEEE, 1999.

[5] O. Kramer, C. Ting, and H. Kleine Büning. A mutation operator for evolution strategies for constrained problems. In *IEEE Congress on Evolutionary Computation, (CEC)*, pages 2600–2606. IEEE, 2005.

[6] A. Ostermeier, A. Gawelczyk, and N. Hansen. Step-size adaptation based on non-local use of selection information. In Y. Davidor and et al., editors, *Parallel Problem Solving from Nature (PPSN)*, pages 189–198. Springer Verlag, 1994.