# An Efficient Evolutionary Programming Algorithm Using Mixed Mutation Operators for Numerical Optimization


Saif Ahmed
Department of Computer Science & Engineering, Bangladesh University of Engineering & Technology
Dhaka-1000, Bangladesh
saif.ahmed@csebuet.org

Md. Tanvir Alam Anik
Department of Computer Science & Engineering, Bangladesh University of Engineering & Technology
Dhaka-1000, Bangladesh
tanviranik@gmail.com

K. M. Rakibul Islam
Department of Computer Science & Engineering, Bangladesh University of Engineering & Technology
Dhaka-1000, Bangladesh
kmrakibulislam@csebuet.org



## ABSTRACT

Evolutionary algorithms often suffer from premature convergence when dealing with complex multi-modal function optimization problems as the fitness landscape may contain numerous local optima. To avoid premature convergence, sufficient amount of genetic diversity within the evolving population needs to be preserved. In this paper we investigate the impact of two different categories of mutation operators on evolutionary programming in an attempt to preserve genetic diversity. Participation of the mutation operators on the evolutionary process is guided by fitness stagnation and localization information of the individuals. A simple experimental analysis has been shown to demonstrate the effectiveness of the proposed scheme over a test-suite of five classical benchmark functions.



## Categories and Subject Descriptors

G.1.6 [**Mathematics of Computing**]: Optimization—*Global Optimization*; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*Heuristic methods*

## General Terms

Algorithms, Performance, Design

## Keywords

Evolutionary Programming, Distribution-Based Mutation, Differential Mutation, Two-Flag Heuristic


## 1. INTRODUCTION

Evolutionary programming (EP), one of the major branches of Evolutionary algorithms (EAs), was first introduced as a paradigm for artificial intelligence. Later, it was extended and successfully applied to many global optimization problems. As EP solely relies on mutation for producing offspring, significant amount of research has been done for designing new mutation operators. For example, conventional EP (CEP) [2] based on Gaussian distribution, Fast EP (FEP) [3] based on Cauchy distribution for mutation. In this paper, we have introduced a mixed mutation scheme for EP based on a two-flag heuristic that controls and ensures proper participation of mutation operators in a convenient way.



*GECCO'13 Companion,* July 6–10, 2013, Amsterdam, The Netherlands.
ACM 978-1-4503-1964-5/13/07.


## 2. METHOD

We have proposed an EP algorithm (FLEP) whose diversity is controlled by **f**itness stagnation and **l**ocalization information of the individuals. The main population is divided into two subpopulations by a division factor (*df*) (initially *df* is set to 0.5 to ensure equal subpopulation size). Subpopulation-1 and Subpopulation-2 are mutated by two different categories of mutation operators whose participations are determined by fitness stagnation and localization information. After a certain number of generations, successful mutation rates of the mutation operators are examined and based on those outcomes, *df* is updated to change the size of Subpopulation-1 or Subpopulation-2.

FLEP differs from some state-of-the-art research works [3], [5], [4] at several points. First, FLEP uses both distribution based (section 2.1) and differential mutation operators (section 2.2) to enhance explorative and exploitative search abilities of EP. Second, FLEP associates a two-flag heuristic to make decisions about the participation of mutation operators. The heuristic depends on two attributes: fitness stagnation and localization. Fitness optimization denotes at what amount an individual is able to optimize its fitness. This attribute is obtained by taking the difference between parent fitness and offspring fitness. If for an individual the fitness optimization is less than a predefined threshold, then fitness stagnation (FS) occurs and the flag for FS is set to 1, otherwise 0. Another attribute for the heuristic is localization information measured by the Euclidian distance between two individuals and observing if it becomes smaller than a certain fraction. If localization occurs for an individual, then the localization flag is set to 1, otherwise 0. The localization methodology implemented in FLEP is to some extent relevant to the concept of niche radius [1]. This attribute signals the presence of duplicate individuals within the population. Thus fitness stagnation and localization attributes together formulate the diversity information of the individuals.

Table 1 shows the two-flag heuristic along with the decision results for choosing desired mutation operator. Here, Random

**Table 1. Two-flag heuristic decision**

| Fitness Stagnation | Localiza-tion | Heuristic Decision for Distribution-based Mutations | Heuristic Decision for Differential Mutations |
|---|---|---|---|
| 0 | 0 | Random | Random |
| 0 | 1 | Cauchy | LNMO |
| 1 | 0 | Gaussian | GNMO |
| 1 | 1 | MMO | DGMO |

means one mutation operator is randomly chosen from corresponding category when both the attributes are 0.

## 2.1 Distribution Based Mutation Operators

It is desirable to adaptively determine the participation of several mutation operators at different stages of evolutionary process. The proposed FLEP scheme combines and ensures the participation of several distribution based mutation operators at the time of their best need. Here, distribution based mutation operators refer to those mutation operators that employ some probability distribution functions to generate the search step size. In this paper, we have chosen one heavy tail Cauchy distribution based Cauchy mutation [3], one short tail Gaussian distribution based Gaussian mutation [2], and mean mutation operator (MMO) [6] whose underlying distribution function is the average of Gaussian and Cauchy distribution. Details of them can be found in [3], [2], and [6] respectively.

## 2.2 Differential Mutation Operators

FLEP also combines and ensures the participation of differential mutation operators as they include neighborhood individuals to formulate diversity information. An individual has two distinct types of neighbors: Local neighbors that are close to the individual considering Euclidean distance and global neighbor that has the best fitness value in the entire population. FLEP uses local neighbor-based mutation operator (LNMO) and global neighbor-based mutation operator (GNMO). Details about them can be found in [4]. Another diversity guided mutation operator (DGMO) is also used by FLEP details of which can be found in [5]. We modified these mutation operators while implementing. For simplicity, we have just referred them and their participation procedure is depicted in Table 1. Also the logical descriptions for their participation are one of the focuses of our future work.

# 3. RESULTS AND DISCUSSION

We have chosen 2 unimodal and 5 multimodal functions from the classical benchmark function set introduced in [3] to present a simple experimental study. Table 2 shows the obtained error results from the experiments for FLEP in comparison with FEP [3]. The error is computed as (Error = $f(x) - f(x^*)$), where $f(x)$ is the obtained solution by the algorithm, while the $f(x^*)$ is the already known global optimum for a particular benchmark function. It is apparent from the table that FLEP achieves excellent optimization performance for both unimodal and multimodal functions. The convergence characteristics for three functions have been presented in Figure 2. It is obvious from the figure that FLEP converges smoothly without getting stuck at local minima until it reaches proximity of global minima. Note that, distribution based mutation operators play a significant role in achieving excellent optimization. To emphasize their importance, we have showed the mean error value obtained by FLEP with $df$=0 (FLEP($df$=0) column in Table 2) which indicates size of Subpopulation-1 is 0 that is all the

**Table 2. Performance on $f_4$, $f_6$, $f_8$, $f_{10}$ & $f_{11}$-$f_{13}$ with number of function evaluations=150000 and dimensions=30(over 25 runs)**

| No | FLEP Mean Error | Std. | FLEP ($df$=0) Mean Error | Std. | FEP Mean Error | Std. |
|---|---|---|---|---|---|---|
| $f_4$ | **9.52e-06** | 2.12e-06 | 5.62e+00 | 1.62e+00 | 3.00e-01 | 5.00e-01 |
| $f_6$ | **0.0** | 0.0 | 3.00e+00 | 0.0 | **0.0** | 0.0 |
| $f_8$ | **4.00e+00** | 0.0 | 21.3333 | 2.36e+00 | 1.50e+01 | 5.26e+01 |
| $f_{10}$ | **4.82e-11** | 1.67e-10 | 3.6416 | 4.70e-01 | 1.80e-02 | 2.10e-03 |
| $f_{11}$ | 3.34e-16 | 0.0 | **2.71e-16** | 5.23e-16 | 1.60e-02 | 2.20e-02 |
| $f_{12}$ | **7.95e-21** | 2.17e-18 | 6.47e+00 | 1.81e-01 | 9.20e-06 | 3.60e-06 |
| $f_{13}$ | **1.84e-20** | 1.05e-19 | 1.10e-02 | 1.72e-11 | 1.60e-04 | 7.30e-05 |

individuals are mutated by differential mutation operators. The results are not satisfactory as evidenced from Table 2.

In summary, the proposed FLEP scheme gets advantage from two different categories of mutation operators whose participation is controlled by a two-flag heuristic. In future, we are interested to explore this idea for a broader class of problems. Also the choice of mutation operators and designing new heuristics for them are attractive research issues and deserve further investigations.

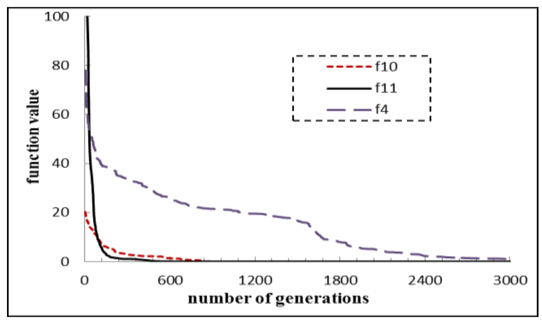


**Figure 1. Convergence characteristics for $f_4$, $f_{10}$ & $f_{11}$.**

# 4. REFERENCES

[1] A. Della Cioppa, C. DeStefano, and A. Marcelli. Where are the niches? Dynamic fitness sharing. *IEEE Trans. Evol. Comput.*, 11(4):453-465, August 2007.

[2] T. Bäck and H.-P. Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary Comput*., 1(1):1–23, 1993.

[3] X. Yao, Y. Liu and G. Lin. Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, 3(2):82-102, July 1999.

[4] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar. Differential evolution using a neighborhood-based mutation operator. *IEEE Trans. Evol. Comput.*, 13(3):526–553, June 2009.

[5] M. S. Alam, M. M. Islam, X. Yao, and K. Murase, "Diversity Guided Evolutionary Programming: A novel approach for continuous optimization," *Appl. Soft Comput.*, vol. 12, no. 6, pp. 1693-1707, Jun. 2012.

[6] K. Chellapilla, "Combining mutation operators in evolutionary programming," *IEEE Trans. Evol. Comput.,* vol. 2, no. 3, pp. 91–96, Sep. 199