

Population Diversity Measurement Methods to Analyze the Behavior of Differential Evolution Algorithm

M.S. Akhila*, C.R. Vidhya** & G. Jeyakumar***

Abstract: Differential Evolution (*DE*), the real parameter optimization algorithm for population based optimization problem, has proved its superiority over variety benchmarking and real time problems. Measuring and visualizing the changes in the diversity of *DE* population during its search is one of the ways to understand the algorithmic behavior of *DE*. This helps to provide better insight for proper tuning of control parameters of *DE*. Hence, an extensive study to describe various possible ways to measure the population diversity of *DE* algorithm would be a useful tool for the researchers and practitioners of *DE*. Towards this research direction, this paper presents variety of population diversity measurement methods available for population based algorithm (in general). As well as, as an initial attempt, three methods out of all the identified methods are implemented for *DE/rand/1/bin* algorithm for a benchmarking function suite with four different functions. The results recorded are presented and discussed in this paper.

Index Terms: Differential Evolution, Variants, Empirical and Theoretical analysis, Diversity in Population, Benchmark Functions, Exploration, Exploitation.

1. INTRODUCTION

The category of algorithms used in Evolutionary Computing (*EC*) field is named as Evolutionary Algorithms (*EAs*). Starting with a random initial possible solutions (initial population) of the given optimization problem, the *EAs* do systematic random search to obtain the global optimum solution. Any evolutionary search process ends successful (reaches the global optimum solution) only if it maintains enough diversity among the candidates of the population. The changes in the diversity of the population are due to the evolutionary processes: exploration and exploitation. At any moment of the search process the algorithm is either in explorative phase or exploitative phase. The exploration phase includes the variation operations: mutation and crossover. The exploitation phase includes the selection operation. Maintaining a balance between these two phases is important (as well as crucial) to preserve the population diversity. By knowing the present phase (explorative or exploitative) of an evolutionary search it is possible to alter the search direction of the algorithm, in order improve its efficiency. The explorative phase of the search adds new candidates to the population and hence increases the population diversity. On the contrary, the exploitative process finds the best candidates among existing candidates, which results decrease in the population diversity. Hence, measuring the population diversity during the search process, at certain frequency, is the way to know the present phase (explorative or exploitative) of search.

This research work aimed at identifying different methods available to measure the population diversity of the population based search algorithms, choosing the methods best suiting for *DE* algorithm and implement them for different variants of *DE* algorithm on a set of benchmarking problems. As a preliminary work,

* Dept of Computer Science and Engineering Amrita School of Engineering, Coimbatore Amrita Vishwa Vidyapeetham Amrita University India, Emails: *akhila251193@gmail.com, **vidhyacr93@gmail.com, ***g_jeyakumar@cb.amrita.edu

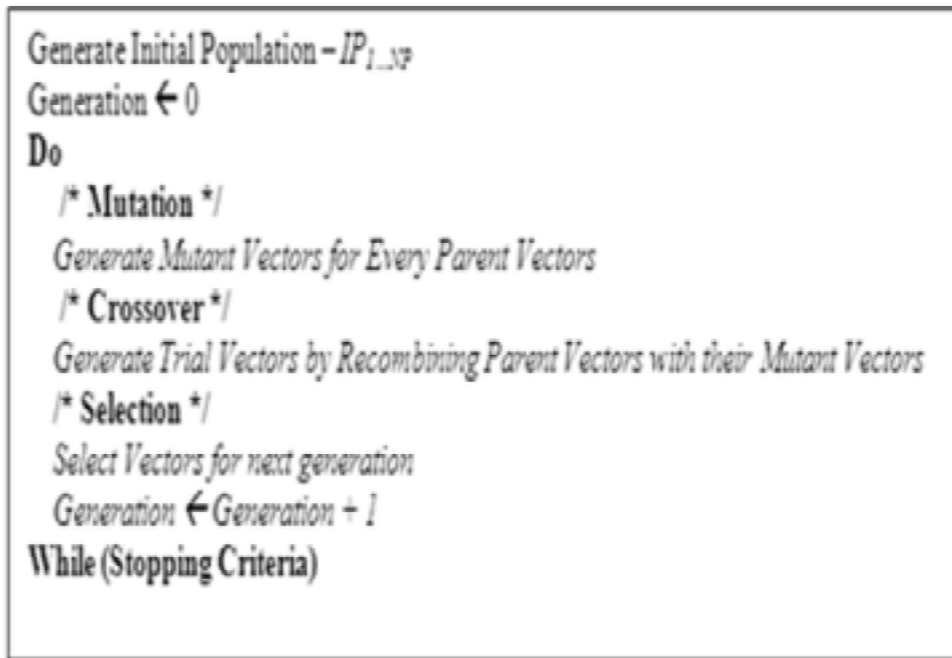


Figure 1. Structure of *DE* algorithm

population of parameter vectors, initialized randomly. The mutation and crossover process of *DE* bring variability in the population, thus regarded as variation operations. The selection process is to direct the search to the optimal solution. A mutant vector is generated for every vector in the current generation by following a specific mutation strategy (there are many mutation strategies available for *DE* algorithm). A common mutation strategy (named and *rand/1*) generates mutant vector by selecting three random vectors from the population and adding the scaled weighted difference of one pair of random vectors to another random vector. Then every vector (parent vectors) in the current generations is recombined with its mutant vector to generate the child vectors (trial vectors). Now, based on the fitness the parent vectors or their trial vectors are selected for the next generation. A simplified algorithmic structure of *DE* is shown in Figure 1.

3. RELATED WORKS

In analyzing the behavior of *DE* algorithm by measuring the population diversity many works are reported in the literature.

The population variance was used to evaluate and analyze the performance of different *DE* variants in [3]. The authors reported insight about the competitiveness of *DE* variants. The reported results say that the best performing variants are showing good convergence speed and explorative power and others fall in premature convergence or stagnation. The competitiveness of the *DE* variants is further analyzed with Convergence Speed and Quality Measure in [4]. The binomial variants are found to outperform other variants in all the cases of functions and parameters. In a recent study reported in [5], it is said that the performance of *DE* can be analyzed using empirical measurement of the population variance. In this work, 14 classical *DE* variants are implemented and suitable evidences are provided for the premature convergence and stagnation cases.

Zaharie [6] proposed a theoretical expression for measuring the expected population variance of *DE*. Zaharie verified the validity of the proposed equation by implementing it for *DE/rand/1/bin* with two benchmarking functions (Rastrigin and Ackley). By extending this idea, in [7] authors derived the theoretical expressions for *DE/rand/2/bin*, *DE/best/1/bin* and *DE/best/2/bin* [7]. In continuation to this work, in [8] the theoretical expression for *DE/current-to-best/1/bin* is derived.

three population diversity measurement methods are implemented for *DE* and the results are presented in this paper.

2. *DE* ALGORITHM

DE was developed by Ken Price's [1, 2] to solve a Polynomial fitting Problem (Chebychev), which is introduced by Rainer Storn. It is capable of handling wide variety of benchmarking and real time optimization problems. Differential Evolution (*DE*) starts its process with a initial

A new population adaptation approach called *APED* (Auto-Enhanced Population Diversity) was proposed by Yang et al. [9], to help the *DE* algorithm to stop the evolution before resulting premature convergence and/or stagnation problem. Daniela Zaharie [10] proposed a diversity enhancing mechanism on the basis of two algorithms related to *DE* algorithms to solve static and dynamic optimization problems. As per her study Zaharie says that premature convergence can be avoided by using multi-population approach and parameter adaptation scheme in static problems. An idea to adapt the parameters of *DE* to control the population diversity was proposed in [11].

Mattiussi et al introduced [12] a diversity measurement mechanism for a population based on the distance metrics. This work was done for a population of strings with variable length defined on a finite alphabet. The semi-metric distances between pairs of strings are calculated for their study.

Generally, a population is considered as diverse if it contains individuals with wide variety of characters. These characters may include demographic and/or cultural characters. A process to measure the diversity of human population based on demographics features is described in [13].

On understanding the availability of various methods to measure the population diversity, this paper aims to implement and show the results of few population diversity measurement methods for *DE* algorithm. Various methods to measure the population diversity are categorized and are explained in section IV.

4. POPULATION DIVERSITY MEASUREMENTS FOR EVOLUTIONARY ALGORITHMS

In general, in all the evolutionary algorithms the search for the global optimum solution is based on the exploration and exploitation processes. An algorithm converges to required solution (global optimum solution), when it preserves a stability between these two processes. When the exploitation process is dominant the population loses its diversity soon and the algorithm falls into a local optimal solution and this is regarded as premature convergence. On the other hand if the exploration process is dominant the algorithm wastes much of its time in exploring uninteresting regions of the search space, this situation makes slow convergence and is called as stagnation. To stop an *EA* falling either in premature convergence or stagnation, the changes in the population diversity is a good indicator. Rapid decrease in the population diversity causes premature convergence and too slow decrease (or increase) causes stagnation. Hence, it becomes mandatory to measure the population diversity of an *EA* during it search to analyze and improve its performance. Based on our survey we identified various methods of measuring population diversity, in common, to any of the population based search algorithms. The identified methods are (shown in Figure 2) categorized in to three types: Category I: By the variation in genetic and morphological features, Category II: By the fitness of the candidates and Category III: By the population demographics.

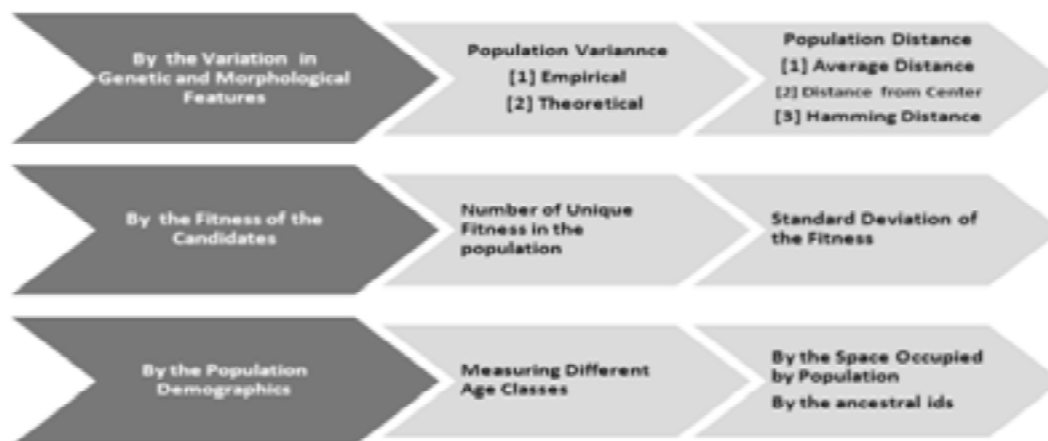


Figure 2: Different Population Diversity Measurement Methods

Each of the categories also includes various methods for population diversity measurements. The methods under Category I use the changes happening to the candidates of the population at the gene/chromosome level due to the variation operations used in the algorithm. The methods in Category II use the quality (fitness) of the candidates in solving the given optimization problem (minimization of maximization) to measure the population diversity. Finally, the Category III methods use the statistical data about the features of the population such as age, space and ancestral data.

This paper is to implement and explain the following three methods chosen from above list for *DE* algorithm

[I] Population Variance

- (1) Empirical variance measurement
- (2) Theoretical variance measurement

[II] Population Distance

- (3) Distance of the best candidate from centre of the population

5. DESIGN OF EXPERIMENTS

The empirical variance of the population is measured as mean of sample variances of the parameters of the population, after the mutation, crossover and selection processes of *DE* algorithm, at the end of every generation.

The theoretical variance of the population is measured by using the equation provided in [6], after the mutation and crossover processes (selection is not considered) of *DE*, at the end of every generation.

The population distance calculation includes finding the average, best candidates of the population and measuring the Euclidean distance between them.

The benchmarking function suite includes functions of different categories: f_1 -Sphere model (unimodal separable), f_2 -Schwefel's Problem (unimodal nonseparable), f_3 -Generalized Restrigin's Function (multimodal

Unimodal Separable Function

f_1 -Sphere mode

$$f_{sp}(x) = \sum_{i=1}^n x_i^2$$

$$-100 \leq x_i \leq 100;$$

$$x^* = (0, 0, \dots, 0);$$

$$f_1(x^*) = 0;$$

Multimodal Separable Function

f_3 -Generalized Restrigin's Function

$$f_3(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$$

$$-5.12 \leq x_i \leq 5.12;$$

$$x^* = (0, 0, \dots, 0);$$

$$f_3(x^*) = 0;$$

Unimodal Nonseparable Function

f_2 - Schwefel's Problem 1.2

$$f_{sch2}(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j \right)^2$$

$$-100 \leq x_i \leq 100;$$

$$x^* = (0, 0, \dots, 0);$$

$$f_2(x^*) = 0;$$

Multimodal Nonseparable Function

f_4 - Ackley's Function

$$f_4(x) = 20 + e - 20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) -$$

$$\exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) \right)$$

$$-5.12 \leq x_i \leq 5.12;$$

$$x^* = (0, 0, \dots, 0);$$

$$f_{4ck}(x^*) = 0;$$

separable) and f_4 -Ackley's Function (multimodal nonseparable). The particulars regarding the definition, range and global solutions of the functions are presented below.

6. RESULTS AND DISCUSSION

Table 1. Empirical Variance Measurement of DE

In many population-based algorithms the diversity in population is measured by the population variance (a population consisting of almost identical elements has low diversity in population). We computed the empirical variance at the component level of the population elements. Since in DE algorithm the transformations are made independently for each parameter of the candidate vectors the empirical variance is computed by averaging the sample variance for all D components. The results recorded for the functions (f_1, f_2, f_3 and f_4) for the 50 generations are presented in Table 1 and Figure 3. It is observed from the results (Table 1 and Figure 3) that $DE/rand/1/bin$ shows gradual decrease in the population diversity for all the functions. This shows the effectiveness of $DE/rand/1/bin$ in solving all the kinds of problems.

G	f_1	f_2	f_3	f_4
1	3407.55	3057.41	8.91	339.24
5	2152.34	2827.63	5.40	221.65
10	1035.70	2500.62	2.95	100.72
15	477.89	2122.48	2.13	50.24
20	194.03	1951.39	1.62	19.50
25	64.36	1549.44	1.34	9.22
30	34.31	1372.22	1.14	3.88
35	15.99	1126.78	0.92	1.68
40	6.36	903.62	0.80	0.88
45	3.00	742.61	0.73	0.43
50	1.29	604.54	0.70	0.23

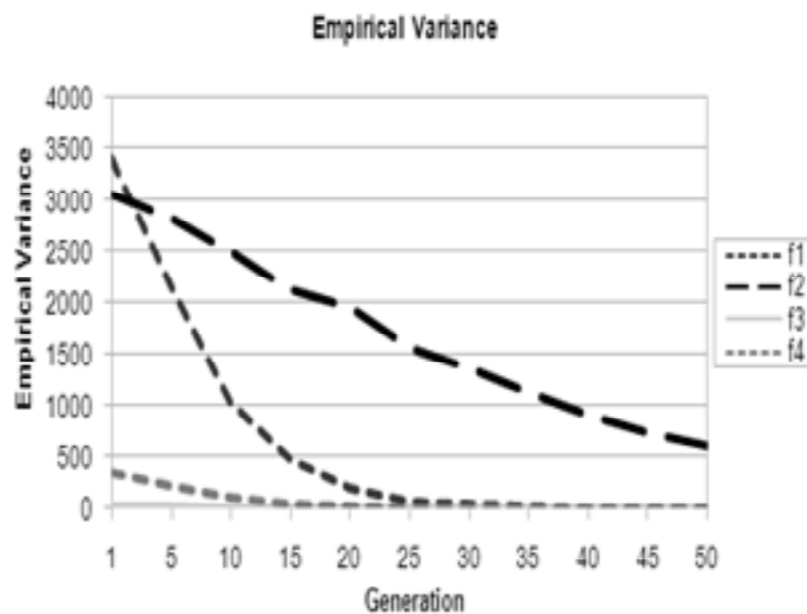


Figure 3: Empirical Variance of DE

Zaharie, in [6], stated a main theorem to relate the initial population variance and the population variance after mutation and crossover. The stated theorems is [6, 7] : “Let $X = \{x_p, \dots, x_m\}$ be the current population, $Y = \{Y_p, \dots, Y_m\}$ the intermediate population obtained after applying the mutation and $Z = \{Z_p, \dots, Z_m\}$ the population obtained by crossing over the population X and Y . If F is the parameter of the mutation step and P_c is the parameter of the crossover step then

$$E(\text{Var}(Z)) = \left(2F^2 p_c + 1 - \frac{2P_c}{m} + p_c^2 \right) \text{Var}(X) \quad (1)$$

Now, the theoretical equation to measure the expected population variance at generation g is

$$E(\text{Var}(X^g)) = \left(2F^2 p_c + 1 - \frac{2P_c}{m} + p_c^2 \right)^g \text{Var}(X^0) \quad (2)$$

where X^0 is the initial population and X^g is the population obtained after g generations.

The theoretical population variances in our experiment are calculated directly by using the equation (2). The theoretical variance measured for all the benchmarking functions are presented in Table 2 and Figure 4. As given in (Zaharie, 2001) the critical value for $F(F_c)$ is calculated from equation (3) for $P_c = 0.5$ and $m = 50$. The calculated F_c value is 0.122474.

$$2F^2 p_c + 1 - \frac{2P_c}{m} + p_c^2 = 1 \quad (3)$$

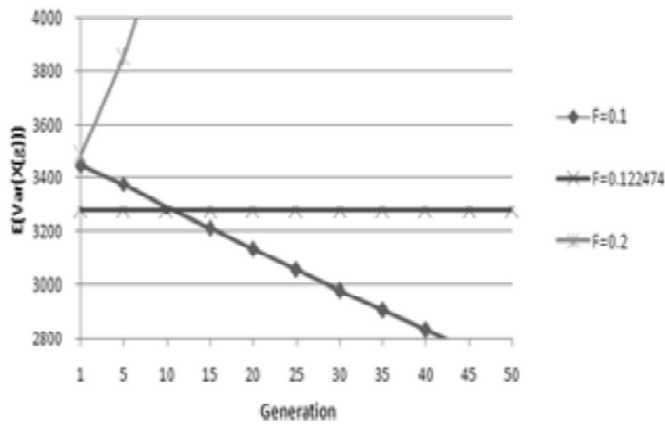
It is interesting to note that (as it is seen from Table 2 and Figure 4), the population variance remains constant when $F = F_c$, decrease when $F < F_c$ and increases when $F > F_c$.

Next the experiment includes calculating the population distance. The population distance is calculated by measuring the Euclidean distance between the best and average candidates of the population. The best candidate is the one with good fitness value in the current population. Since all our benchmarking functions used in our experiment are minimization functions, we selected the candidate with least objective function value at each generation as the best candidate. The average candidate is a newly generated candidate with its parameter values as the average of the parameter values of all the candidates in the

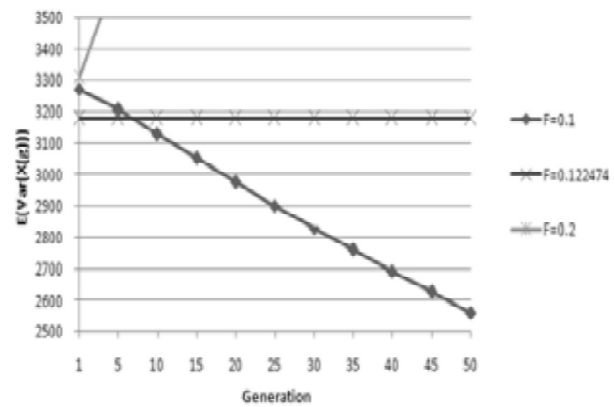
Table 2
Theoretical Variances Measured for DE Algorithm

G	f_1			f_2		
	$F = 0.1$	$F = 0.12$	$F = 0.2$	$F = 0.1$	$F = 0.12$	$F = 0.2$
1	3443.25	3278.58	3484.94	3272.53	3176.51	3307.86
5	3374.90	3278.58	3846.72	3207.57	3176.51	3651.26
10	3291.37	3278.58	4352.21	3128.18	3176.51	4131.06
15	3209.90	3278.58	4924.12	3050.75	3176.51	4673.92
20	3130.45	3278.58	5571.20	2975.24	3176.51	5288.11
25	3052.97	3278.58	6303.30	2901.60	3176.51	5983.01
30	2977.41	3278.58	7131.60	2829.78	3176.51	6769.22
35	2903.71	3278.58	8068.75	2759.74	3176.51	7658.76
40	2831.84	3278.58	9129.05	2691.44	3176.51	8665.18
45	2761.75	3278.58	10328.68	2624.82	3176.51	9803.85
50	2693.39	3278.58	11685.96	2559.85	3176.51	11092.16

G	f_3			f_4		
	$F = 0.1$	$F = 0.12$	$F = 0.2$	$F = 0.1$	$F = 0.12$	$F = 0.2$
1	9.30	9.31	8.89	347.71	323.97	356.24
5	9.11	9.31	9.81	340.81	323.97	393.22
10	8.89	9.31	11.10	332.37	323.97	444.90
15	8.67	9.31	12.56	324.15	323.97	503.36
20	8.45	9.31	14.21	316.12	323.97	569.51
25	8.24	9.31	16.08	308.30	323.97	644.34
30	8.04	9.31	18.20	300.67	323.97	729.01
35	7.84	9.31	20.59	293.23	323.97	824.81
40	7.65	9.31	23.29	285.97	323.97	933.20
45	7.46	9.31	26.35	278.89	323.97	1055.83
50	7.27	9.31	29.82	271.99	323.97	1194.58



(a)



(b)

Figure 4: The Evolution of Theoretical Variances of DE (f_1 – (a) and f_2 – (b))

Table 4
The population distance measured for DE algorithm

G	f_1	f_2	f_3	f_4
1	104.85	189.25	6.82	31.16
5	126.03	130.46	4.75	32.51
10	154.97	175.93	6.19	47.51
15	150.49	156.89	7.18	39.81
20	182.16	101.57	8.68	47.56
25	181.95	136.77	10.72	50.19
30	209.25	137.34	8.65	60.68
35	201.05	147.34	9.76	57.63
40	194.04	156.56	9.71	72.90
45	217.11	159.22	10.04	77.86
50	219.92	176.96	9.95	68.00

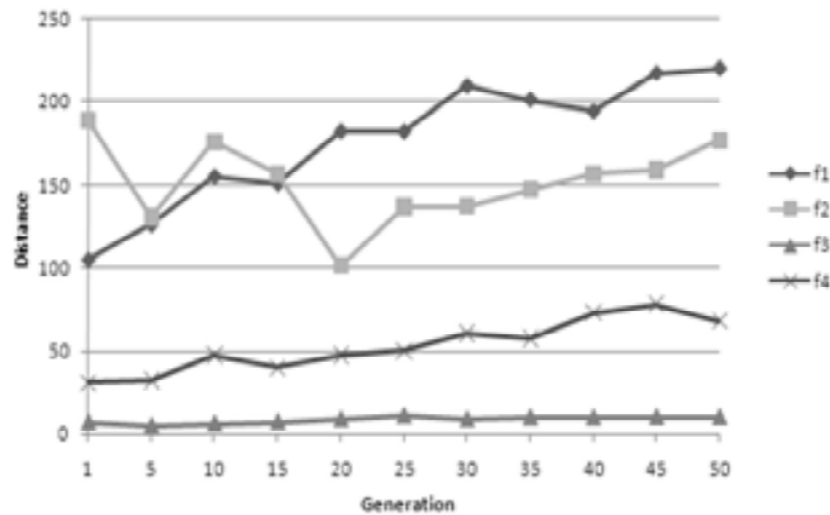


Figure 5. The population distance for *DE* algorithm

population of the current generation. The population distance measured for all the test functions are presented in Table 4 and Figure 5. As a difference, we observed from the results that there is no proper pattern of increasing or decreasing in the values of population distance. It is seen from the Figure 5 that the population distance is increasing at generation for all the functions but not at uniform or consistent rate, it sometimes decreases also.

Thus the paper shown the results obtained for measuring the population diversity of *DE* algorithm by using three different strategies. Even though no proper inference is derived out of the results shown in this paper, this has formed a base for our further research to implement other methods of measuring population diversity (shown in Figure 2) and deriving required information to alter the *DE* search (in improving its performance).

7. CONCLUSION

In this paper, we explained measuring population diversity empirically, theoretically and using the distance of the best candidate from the center of the population. A *DE* variant (*DE/rand/1/bin*) is used to implement and test the identified population diversity methods, for different test functions. The results showed that the empirical population variance is decreasing at slow rate for all the test functions. The theoretical variances are increasing, decreasing or remains constant for varying values of F . Finally, the population distance measurement was found to have irregular pattern of increase or decrease. This work can be extended further to analyze the explorative/exploitative behavior of *DE* algorithm by implementing other population diversity measurement methods.

References

- [1] R. Storn, and K. Price, "Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces", *Technical Report-TR-95-012*, 1995.
- [2] R. Storn, and K. Price, "Differential evolution—a simple and efficient heuristic strategy for global optimization and continuous spaces", *Journal of Global Optimization*, Vol. 11, No. 4, pp. 341–359, 1997.
- [3] G. Jeyakumar, and C. Shanmuga Velayutham, "Analyzing the explorative power of differential evolution variants on different classes of problems", *Lecture Notes in Computer Science (LNCS-6466)*, Springer-Verlag, pp. 95-102, 2010.
- [4] G. Jeyakumar, and C. Shanmuga Velayutham, "Convergence Analysis of Differential Evolution Variants of Unconstrained Global Optimization Functions", *International Journal of Artificial Intelligence & Applications*, vol. 2, No. 2, pp. 116 – 127, 2011.

- [5] S. Thangavelu, G. Jeyakumar, and C. Shunmuga Velayutham, "Population Variance Based Empirical Analysis of the Behavior of Differential Evolution Variants" *Applied Mathematical Sciences*, Vol. 9, No. 66, pp. 3249-3263, 2015.
- [6] D. Zaharie, "On the explorative power of differential evolution algorithms", *Proceeding-3rd International Workshop on Symbolic and Numeric Algorithms on Scientific Computing*, SYNASC-2001, 2001.
- [7] G. Jeyakumar, and C. Shunmuga Velayutham, "A comparative study on theoretical and empirical evolution of population variance of differential evolution variants", *Lecture Notes in Computer Science (LNCS-6457)*, pp. 75-79, 2010.
- [8] S. Thangavelu, G. Jeyakumar, Roshni M. Balakrishnan, and C. Shunmuga Velayutham, "Theoretical Analysis of Expected Population Variance Evolution for a Differential Evolution Variant", *Computational Intelligence in Data Mining*, Springer, Vol. 2, pp. 403-416, 2015.
- [9] M. Yang, C. Li, Z. Cai, and J. Guan, "Differential evolution with auto-enhanced population diversity", *IEEE Transactions on Cybernetics*, Vol. 45, No. 2, pp. 302-315, 2015.
- [10] D. Zaharie, and F. Zamfirache, "Diversity enhancing mechanisms for evolutionary optimization in static and dynamic environments", *In 3rd Romanian-Hungarian Joint Symposium on Applied Computational Intelligence*, pp. 460-471, 2006.
- [11] D. Zaharie, "Control of population diversity and adaptation in differential evolution algorithms". *In Proceedings of MENDEL*, Vol. 9, pp. 41-46, 2003.
- [12] C. Mattiussi, M. Waibel, and D. Floreano, "Measures of diversity for populations and distances between individuals with highly reorganizable genomes", *Evolutionary Computation*, Vol. 12, No. 4, pp. 495-515, 2004.
- [13] R. L. Summers, L. Jackson, L.H. Woodward, J. Jones, and S. Dryer, "Methodology for measuring diversity in human populations", *Journal of Cultural Diversity*, Vol. 18, No. 2, pp. 43-47, 2011.