International Journal of Technical Innovation in Modern Engineering & Science (IJTIMES)
Impact Factor: 5.22 (SJIF-2017), e-ISSN: 2455-2585
Volume 4, Issue 10, October-2018

OPTIMIZATION PROBLEMS SOLVED BY DIFFERENTIAL EVOLUTION ALGORITHM

Hitesh Sharma[1]

[1]*Computer science department,Pratap Institute Of Technology & Science ,Sikar,*

***Abstract*— In recent years differential evolution is most important method for solving computer science and mathematical optimization problem. This type of optimization problems generally solved by metaheuristic procedure. Metaheuristic is a higher level procedure to find, generate or select heuristic that may provide sufficiently good solution to an optimization problem. Metaheuristic procedure is used to obtain attractive result in solving much higher degree equation and engineering optimization problems. Differential evolution has two major problems. First, its performance is significantly influenced by the control parameters, which are problem dependant and which vary in different regions of space exploration. Secondly, it can easily get stuck in a local optimum or fail to generate better solutions before the population has converged.**

This research paper aims to develop new differential evolution algorithms to address the two mentioned problems. The methods proposed in this research paper have been tested in different benchmark problems. All the result of this research paper is compare with the state of the art algorithms and other competitive result.

***Keywords*— CrossoverRate , EvolutionaryAlgorithm , mutationFactor , LocalSearch**

I. INTRODUCTION

Nowadays, industries face many optimization problems in their system and product development. Optimization algorithms help in reducing the cost of industrial product and also enhance the performance of product. In the industrial world, companies often need to optimize one or more factor (less cost, better performance, etc.), where optimization can play an important role, adjusting of fine tuning system design in terms of factors mentioned above. When we solve different problems with different characteristics then metaheuristics is good method for finding good and acceptable solution in proper time duration compared with traditional algorithms .metaheuristics method of optimization can be classify into two categories : first is population based approach and second is trajectory based approach . These two categories of metaheuristics method of optimization is categorised in different sectors like as hill-climbing, simplex method variable neighbourhood search etc. Approaches are considered in trajectory based algorithms and examples of population based approaches are group of solution are handled at same time, real world optimization problems etc.

Differential Evolution hybridization with local search algorithms is new topic in computer science research work. Local search algorithms method can bring two benefits. First is enhance the exploitation of some research areas and second is they can drive the search into a better area further from local optimization. Differential evolution approach is popular in many area due to its high ability, simple implementation and easy use . This research paper aims to develop new technique and method to mitigate the two aforementioned problems of deferential evolution in solving typical optimization problem.

II. DIFFERENTIAL EVOLUTION

Differential evolution for optimization aim to create continuously improved approximation to the problem in a number of treks. Metaheuristics method is divided into two categories and each categories have special characteristic. In relevance to the research of the paper , differential evolution algorithm that belongs to the population-based and Alopex search is belong to the trajectory-based .the population is composed of Np individuals and each individual in population represents a possible solution to the problem . Differential evolution works in three steps

1. Mutation
2. Crossover
3. Selection

Mutation :- A vector for the mutated solution is called mutant vector .Np fluctuate individuals are generated using some individuals of the population . if we notated differential evolution as DE then differential evolution is notated in mutation is DE/x/y. where x stands for the vector to be mutated and y represents the number of difference vector used in the mutation.

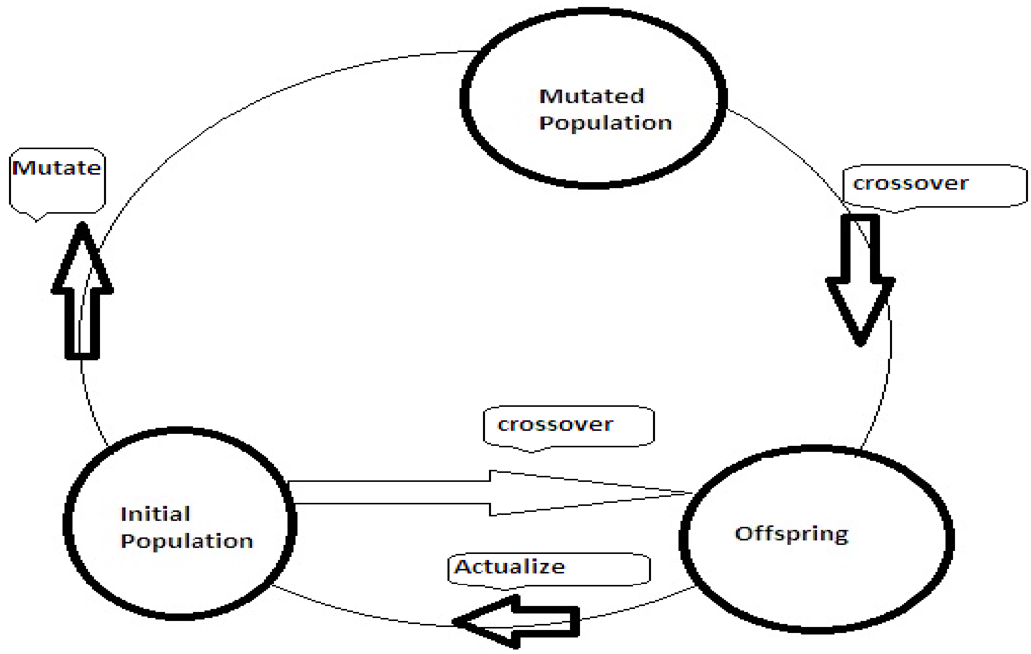

Fig. 1 Differential Evolution Process

Mutant vector is created according to Eq.

$$V_{i,g} = X_{r1,g} + F\,(X_{r2,g} - X_{r3,g})$$

where $V_{i,g}$ represents the mutant vector.
i stands for the index of the vector.
g stands for the generation.

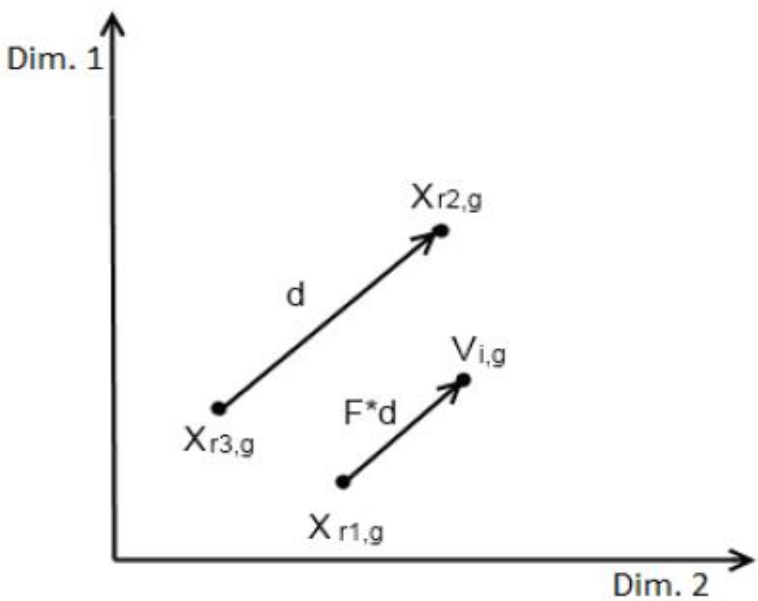

Fig. 2 Random mutation with one difference vector

III. CROSSOVER :-

In crossover, we recombine the set of mutant vector created in the first stage with the original population members to generate offspring solutions. In crossover we denote offspring by $O_{i,g}$ where i stands for index and g stands for generation . Crossover is classified in two basic strategies .first is exponential and other is binomial.

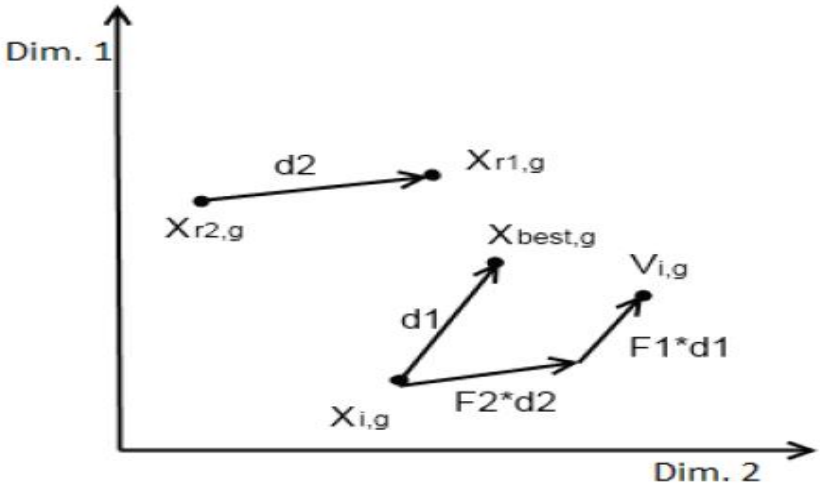

Fig. 3 Mutation with one difference vector

Where k stands for the index of every parameter in a vector and CR is the probability of the recombination.

Binomial crossover example shown below

$$O_{i,g}[k] = \begin{cases} V_{i,g}[k] & \text{if } (0,1) < CR \\ X_{i,g}[k] & \text{otherwise} \end{cases}$$

Vi,g	6.5	8.1	4.3	6.3	3	6.1	3
Xi,g	1.7	7	2.4	6.1	10	1	5.7

Binomial Crossover

Oi,g	6.5	8.1	2.4	6.3	10	6.1	5.7

The offspring vector is derived in equation

$$O_{i,g}[k] = \begin{cases} V_{i,g}[k] & \text{if } K \in \{ m, (m+1)n, \text{- - - - - - - - - - - -} (m+p-1)n\} \\ X_{i,g}[k] & \text{otherwise} \end{cases}$$

Where m and p are two randomly selected integers.

Exponential crossover example shown below

Vi,g	6.5	8.1	4.3	6.3	3	6.1	3
Xi,g	1.7	7	2.4	6.1	10	1	5.7

Exponential

Crossover

Oi,g	1.7	7	4.3	6.3	3	1	5.7

IV. SELECTION

In this step offspring is compared with its parent and stronger one will enter the population for the next generation .the next generation is created according to given below equation.

$$X_{i,g=1} = \begin{cases} Q_{i,g} & \text{if } f(Q_{i,g})<f(X_{i,g}) \\ X_{i,g} & \text{otherwise} \end{cases}$$

Where
$X_{i,g}$ is an individual in the old population .
Xi,g+1 is the individual in the next generation .
Oi,g is the offspring .
f stands for the objective function of the optimization problem .

V. LOCAL SEARCH IN DIFFERENTIAL EVOLUTION

Local search technique can be divided into two groups according to Lozano et al.

1. Crossover – based local search :- in this group the local search is implemented by recombination operators in which the individuals from the population are used as parents to produce offspring around them.
2. Local improvement process oriented local search:- Local search heuristics in this group applies local improvement on one or more individuals in the populations.

VI. LOCAL SEARCH IN DIFFERENTIAL EVOLUTION

When engineers and researches use local search in differential evolution then they use different strategies . there are some hybridization that perform local search after a certain number of generation. Two different approaches that follow this strategy. First is update the population after performing the local search and second is the population is not update after local search.

VII. CONCLUSION

1. Enhance the search power of differential evolution.
2. Greedy adaptive differential evolution algorithm has been developed.
3. Improve the performance of differential evolution.
4. Practical industrial problem solved by differential evolution with accurate result.
5. Greedy adaptive differential evolution should be attractive in engineering applications.

REFERENCES

[1] J.H. Van Sickel, K.Y. Lee, and J.S. Heo. Differential evolution and its applications to power plant control. In International Conference on In-telligent Systems Applications to Power Systems, 2007.
[2] R. Helena, O.M. Lourenzo, and T. Stutzle. Iterated local search. Springer, 2003.
[3] S. Kirkpatrick, C.D. Gelantt, and M.P. Vecchi. Optimization by simulated annealing. Science, 220:671–680, 1983.
[4] M.Avriel.Nolinear programming: Analysis and methods .In Dover Publishing, 2003.
[5] T.A.Feo and M.G.C. Resende. Greedy ran random adaptive search pro-cedures. Journal of Global Optimization, 6:109–134,1995.
[6] R.Ruiz and T.Stutzle. A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. European Journal of Operational Research,177(3):2033–2049,2007.
[7] Gu Jirong and Gu Guojun. Differential evolution with a local search operator. In Proc.2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics(CAR),Wuhan,China,volume2,pages480–483,2010.
[8] D.Molina, M.Lozano, C.Garcia-Martinez, and F. Herrera. Memetic algorithms for continuous optimization based on local search chains.Evo-lutionary Computation,18:27–63,2010.
[9] E.Harth and E.Tzanakou. Alopex: A stochastic method for determining visual receptive fields. Vision Research ,14:1475–1482,1974.
[10] D.Zaharie. Influence of crossover on the behavior of differential evolution algorithms. Applied Soft Computing, 9:1126–1138,2009.
[11] Q.fan and X.Yan. Self adaptive differential evolution algorithm with discrete mutation control parameters. Expert Systems with Applications,42:1551–1572,2015.