A New Scale Factor for Differential Evolution Optimization

Devendra Tayal
Associate Professor & H.O.D
Indira Gandhi Institute of Technology, Delhi

Charu Gupta
Assistant Professor
Bhagwan Parshuram Institute of Tech., Delhi

ABSTRACT
In this paper, we propose a new scale factor in differential evolution for optimization of numerical data (low dimensional data) that is both seen in algebraic and exponential form in real world scenarios. With the present work we improve the optimization of DE with real world numerical data set of the Lahi crop production of Pantnagar farm, G.B. Pant University of Agriculture and Technology, Pantnagar, India; inventory demand and population of India. This study focusses on optimization of numerical data that is characterized by single dimension.

Keywords
Differential evolution (DE), scale factor, optimization, numerical data.

1. INTRODUCTION
Optimization is a very important phase of any data processing model. Since its inception in 1995, DE has drawn the attention of many researchers all over the world resulting in a lot of variants of the basic algorithm with improved performance. Differential Evolution is one of the most powerful stochastic real- valued optimization algorithms in current use. Unlike traditional evolutionary algorithms, Differential Evolution variants induce variety by scaling the differences of randomly selected population vectors and do not use any separate probability distribution for generating offspring. [1]

Optimization computes error value as

True Value – Approximate value

In order to obtain the correct measure of accuracy in an approximate solution, bound of either of the following is obtained:

$$\text{Relative Error} = \frac{|\text{Error}|}{\text{True Value}}$$

$$\text{Absolute Error} = |\text{Error}|$$

Optimization is the study of problems that use numerical approximation for the problems of mathematical analysis. In many practical applications analytical methods are unable to give desirable results. In such cases, we often ignore the exact solution and uses approximate methods instead. Thus, the aim is to give efficient methods to obtain answers of such problems in numerical form.

The optimization problem that is considered here is in the following form, where a local minimum x* is defined as a point for which there exists some $\delta > 0$ so that for all x such that

$$\| x - x^* \| \le \delta;$$

the expression

$$f(x^*) \le f(x)$$

holds on some region around x* all of the function values are greater than or equal to the value at that point.

The motivation behind the present work is to study the convergence of Differential Evolution for the optimization of low dimensional numerical data. In the present paper we discuss Differential Evolution as a numerical optimization technique with improved scale factor that helps in inducing variety and selection of pseudo-random population vector that affects the convergence.

The paper has been organized as follows: Section II discusses the Differential Evolution as an optimization algorithm. Section III explains the proposed scale factor and working methodologies of differential evolution with the new proposed scale factor. Section IV discusses the experimental results and finally in section V we conclude the results with future work.

2. DIFFERENTIAL EVOLUTION
Differential evolution (DE) is a simple evolutionary algorithm that was first introduced by Storn & Price in [10] and was primarily developed to optimize real parameter, real valued functions. From then on DE has emerged as a strong, robust, simple yet effective optimization technique. It is a population-based stochastic technique that is suitable for problems where objective functions are non-linear, non-differentiable, non-continuous, noisy, flat, multi-dimensional or have many local minima, constraints or stochasticity [2]. Such problems are difficult to solve analytically. In such cases differential evolution can be used to find approximate solutions to problems. DE is a simple real parameter optimization algorithm that works in cycles as shown in Fig. 1.

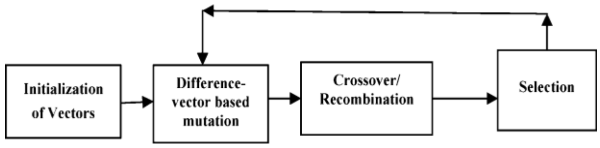

Fig. 1: Main Stages of Differential Evolution

DE is a population based stochastic technique that searches for a point in D-dimensional real parameter space. It begins with randomly initializing the population vectors. Each of the individual vectors is a chromosome. The characteristic of DE is that it constructs, at every generation, for each element of the given population, a mutant vector. The mutant vector is constructed through a mutation operation based on adding differences between randomly selected elements of the same population to another element[2]. For instance, in classical DE, a mutant vector y is constructed from a current population {x1, x2, x3….xm) in the following manner:

$y = x_{r1} + F * (x_{r2} - x_{r3})$ where xr1 , xr2 and xr3 are distinct random indices selected from the current population {1, ….., m} where F is a scalar factor usually $\in$ [0,1].

Based on the mutant vector, a trial vector is constructed through a crossover operation that combines the components of the present vector and the mutant vector, according to a control parameter Cr $\in$ [0,1] called crossover rate. The trial vector is compared with the current population element and the best one, with respect to the objective function, is admitted to the next generation. In the similar way, the mutant vector is generated for every individual and is compared with the trial vector for n number of iterations.

The determination of the number of iterations is application dependent [5, 7, 8]. As explained, the mutant vector is generated by scaling the difference of two random vectors and adding it to another random vector. The scaling is done with a scale factor F. In the present work we propose a new scale factor for optimization of numerical data. The proposed scale factor aims at showing better convergence of DE with low dimensional data. It is also interesting to study the variation in optimization with extreme values, i.e, for outliers [11, 12].

3. PROPOSED SCALE FACTOR

Biologically, "mutation" means a change in the characteristics of a chromosome. In the context of DE, however, mutation is seen as a perturbation with a random element. Actually it is the process of mutation that demarcates one DE scheme from another. The variants of DE are well studied in [2,3]

The proposed variation in scale factor is controlled by a random number (rand) in the range [0, 1] .The scale factor Fx is varied in a random manner that is determined by using the relation:

$$F_x = F_{mean} * \text{rand}(0,1) \qquad (1)$$

where, Fx is the value of the scale factor F at ith year, rand random number in the range [0,1] and Fmean is the mean value of the scale factor usually set to 0.75 for normal iterations [7,8]. For the sake of clarity, the working of DE is explained in the pseudo-code in Fig. 2.

Initialize population P = { $x_1, x_2, x_3, \ldots, .x_m$}, $x_i \in$ D
repeat
for i=1 to m do
Compute scaling factor for i^{th} individual

$$F_x = 0.5 * \text{rand}(0,1)$$

generate a new mutant vector *y*

$$y = x_{r1} + F_x * (x_{r2} - x_{r3})$$

If f(y) < f (x) then insert y into new generation
else insert x into new generation
End if
End for
Until stopping criteria

Fig. 2: Improved Differential Evolution: Algorithm in pseudocode

4. RESULTS AND EXPERIMENTS

DE with the proposed scale factor has been applied to optimize a real world data set of Lahi crop production. The historical data of Lahi Production is in terms of productivity in Kilogram per hectare as shown in Table 1 [9].

Table 1: Historical Lahi Crop Production [8]

Year	Production (Kg/Ha)
1981	1025
1982	512
1983	1005
1984	852
1985	440
1986	502
1987	775
1988	465
1989	795
1990	970
1991	742
1992	635
1993	994
1994	759
1995	883
1996	599
1997	499
1998	590
1999	911
2000	862
2001	801
2002	1067
2003	917

The results of the optimization with the new scale factor for Lahi production from the year 1982 to 2003 is shown in Table 2. In the following, we compute the mean square error (MSE) to observe the convergence of DE with the proposed scale factor. The MSE is given in (2) where Actual Valuei denotes the actual value and Optimized Valuei denotes the optimized value of year (i), respectively.

$$\text{MSE} = \frac{\sum_{i=1}^{n} (\text{Actual Value}_i - \text{Optimized Value}_i)^2}{n}$$

Table 2: Optimized Forecasted Values

Population production		Optimized Lahi	
At G=0			
X1(0) =	[704]		604
X2(0) =	[1030]		1030.1
X3(0) =	[704]		704.1
X4(0) =	[453]		452.92
X5(0) =	[507]		506.5
X6(0) =	[785]		785.29
X7(0) =	[482]		482.3
X8(0) =	[798]		798.02
X9(0) =	[988]		988.1
X10(0) =	[380]		380
X11(0) =	[761]		761
X12(0) =	[988]		987.5
X13(0) =	[380]		379.7
X14(0) =	[761]		760.5
X15(0) =	[772]		771.8
X16(0) =	[501]		500.6
X17(0) =	[595]		595.0
X18(0) =	[887]		886.5
X19(0) =	[772]		772.1
X20(0) =	[772]		772.2
X21(0) =	[1061]		1061.18
X22(0) =	[944]		943.8
X23(0) =	[944]		943.7

The obtained mean square error value for lahi production (eq. 2) is 4.347 and thus the accuracy of optimization is nearly 90%. From Table 2 it can be observed that Differential Evolution with the proposed scale factor has successfully optimized all the individual values except for a single value that shows a deviation of 10% i.e. has an error rate of less than 5%. All other values have successfully been optimized with the proposed scale factor.

The proposed scale factor has been studied on other real world datasets of inventory demand [4] and on the population data of India from year 1930-2000 [6]. The optimized result for these data sets has been provided in Table 3 and Table 4 along with the computed mean square error rate.

Table 3: MSE for Inventory Demand Data Set[3]

Time	Actual Demand	Proposed
20	**227**	**222**
21	**223**	**224**
22	**242**	**238**
23	**239**	**235**
24	**266**	**265**
MSE		**10.01**

Table 4: Computed Mean Square Error (MSE) For Population of India (1930-2000)

Year	Population of India	Optimized population (India)	Year	Population of India	Optimized population (India)
1930	277175	--	1967	504345	**504345**
1931	279115	**279115**	1968	515601	**515601**
1932	284102	**284102**	1969	527177	**527177**
1933	287902	**287902**	1970	539075	**539075**
1934	291753	**291753**	1971	547900	**547900**
1935	295666	**295666**	1972	563530	**563530**
1936	299614	**299614**	1973	575887	**575887**
1937	303626	**303626**	1974	588299	**588299**
1938	307694	**307694**	1975	600763	**600763**
1939	311820	**311820**	1976	613273	**613273**
1940	316004	**316004**	1977	630200	**630200**
1941	318826	**318826**	1978	644330	**644330**
1942	324180	**324180**	1979	658730	**658730**
1943	328255	**328252**	1980	688956	**688956**
1944	332332	**332332**	1981	685200	**685200**
1945	336562	**336562**	1982	703570	**703570**
1946	340796	**340796**	1983	719090	**719090**
1947	345085	**345085**	1984	734870	**734870**
1948	349430	**349430**	1985	749184	**749184**
1949	353832	**353832**	1986	767200	**767200**
1950	350445	**350445**	1987	783730	**783730**
1951	363211	**363211**	1988	797526	**797526**
1952	369231	**369231**	1989	817490	**817490**
1953	375633	**375633**	1990	833929	**833929**
1954	382438	**382438**	1991	843931	**843931**
1955	395096	**395096**	1992	883473	**883473**
1956	397334	**397334**	1993	900453	**900453**
1957	405450	**405450**	1994	918570	**918570**
1958	414021	**414021**	1995	934228	**934228**
1959	423053	**423053**	1996	945121	**945121**
1960	431463	**431463**	1997	962378	**962378**
1961	438800	**438800**	1998	979673	**979673**
1962	452378	**452378**	1999	997515	**997515**
1963	462196	**462196**	2000	1014003	**1013854**
1964	472305	**472305**			
1965	482706	**482706**	**MSE**		**317**
1966	493389	**493389**			

The experimental results in Table 3 & 4 show that the proposed scale factor gets reasonably good optimization results for inventory demand dataset and population data set. This indicates that huge data values are not a matter of concern when it comes to optimization with Differential evolution.

It can be observed from Fig. 4 that the proposed scale factor scales well for the numerical data values. Fig. 5 shows the curve of optimized population values obtained. The overlapping linear behavior indicates a higher optimization. The optimized population is plotted against the actual population (India) data from year 1930 to 2000. A concave upward graph indicates a higher fitness of the proposed scale factor.

5. CONCLUSION AND FUTURE WORK

In this paper, we have proposed a new scale factor in Differential Evolution Algorithm. The algorithm with the proposed scale factor is applied to optimize Lahi production data. Fig. 3 shows the comparison of actual Lahi production with the optimized production. We can clearly observe that the algorithm is efficient and is able to optimize numerical data with a greater accuracy. The approach has also been tested for other real world numerical data set of inventory demand [4], population of India [6] and the optimization accuracy can be seen in Fig. 4 and 5 respectively. The proposed scale factor is able to find the optimal solution without getting "stuck" in local optima. It has also handled the extreme values in an elegant manner by randomizing the process during the creation of mutant vector. We have used the scale factor for low dimensional numerical data; the approach can be further extended to optimize n-dimensional data and can be investigated more deeply for n- parameter problem. The behavior can be analyzed for extreme values i.e. for the outliers, in the datasets. The technique can also be used for non-numerical data which constitutes multi criteria decision making.

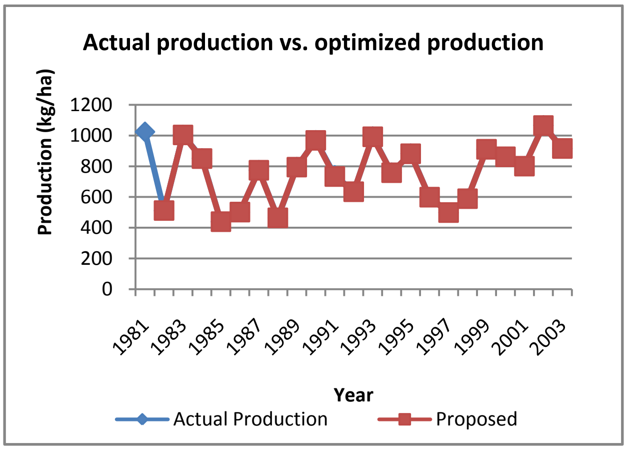

Fig. 3: Actual production vs. optimized production

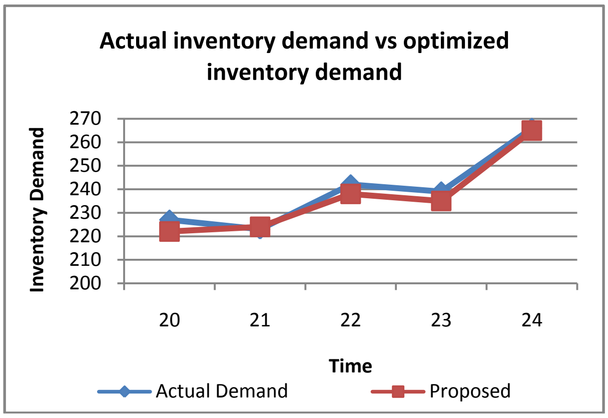

Fig. 4: Actual inventory demand vs. optimized inventory demand

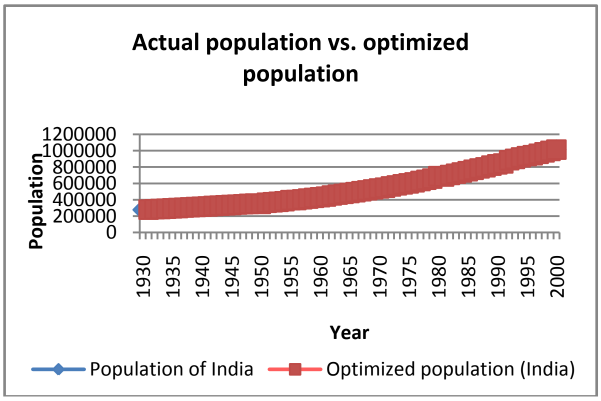

Fig. 5: Population of India vs. optimized population (India)

6. REFERENCES

[1] Ardia D., Boudt K., Carl P., Mullen K.M. and Peterson B.G. 2011. Differential Evolution with DEoptim, *The R Journal Vol. 3(1).*

[2] Das, S., Abraham, A. and Konar, A., 2008a. Automatic clustering using an improved differential evolution algorithm. *IEEE Transactions on Systems, Man and Cybernetics- Part A: Systems and Humans*, 38(1).

[3] Das S., Abraham A. and Konar A., 2008b. Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspective. *Studies in Computational Intelligence, Springer*, 116, 1-38.

[4] Huarng, K. and Yu, H.K., 2006. Ratio-based lengths of intervals to improve fuzzy time series forecasting. *IEEE Transactions on systems, man, and cybernetics—Part B: cybernetics*, 36(2), 328–340.

[5] Holland, J.H., 1975. *Adaptation in Natural and Artificial Systems.* MI: University of Michigan Press, Ann Arbor.

[6] Lahmeyer, J.J., 2003. *India, Historical Demographical data of the whole country.* [Online] Available at: http://www.populstat.info/ [accessed August 2011]

[7] Paterlini, S. and Krink, T., 2006. Differential evolution and particle swarm optimization. *Computational Statistics & Data analysis*, 1220-1247.

[8] Paterlini S. and Minerva T., 2003. Evolutionary approaches for cluster analysis. *In soft computing applications, A. Bonarini, F. Masulli and G. Pasi, Eds. Berlin, Germany: Springer-Verlag,* pp.167-178.

[9] Singh, S. R., 2008. A computational method of forecasting based on fuzzy time series. *Mathematics and Computers in Simulation,* 79, 539–554.

[10] Storn, R. and Price, K., 1997. Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Space. *Journal of Global Optimization,* 341–359.

[11] Qin A.K., Huang V.L. and Suganthan P.N., 2009. Differential Evolution Algorithm with Strategy Adaptation for Global Numerical Optimization. *IEEE Transactions on evolutionary computation, 13(12).*

[12] Mininno, E.; Neri, F.; Cupertino, F.; Naso, D. 2011, "Compact Differential Evolution," Evolutionary Computation, *IEEE Transactions on, vol.15, no.1,* pp.32-54.