Influence of Weighting factor and Crossover constant on the behavior of Differential Evolution Algorithms with a Penalty Function approach

Conference Paper · September 2018

2 authors:

Rasim Salkoski
University for Information Science and Technology "St. Paul the Apostle"

Ivan Chorbev
Ss. Cyril and Methodius University in Skopje

Influence of Weighting factor and Crossover constant on the behavior of Differential Evolution Algorithms with a Penalty Function approach

Rasim Salkoski[1], Ivan Chorbev[2]

[1] University for Information Science and Technology, Building at ARM, 6000 Ohrid, R. of Macedonia
`rasim.salkoski@uist.edu.mk`

[2] Faculty of computer science and engineering, University of Ss Cyril and Methodius, "Rugjer Boshkovikj" 16, P.O. Box 393, 1000 Skopje,R. of Macedonia
`ivan.chorbev@finki.ukim.mk`

Abstract. Differential Evolution (DE) is one of the most popular evolutionary optimization technique on continuous domains based on simplicity, effectiveness and robustness. The weighting factor(F) and crossover constant(CR) allows the construction of a new trial element based on the current and mutant elements. The crossover constant controls which and how many components are mutated in each element of the current population. The work in the present paper aims to analyze the impact the weighting factor and the crossover constant, has on the behavior of DE. The influence of the crossover constant on the distribution of the number of mutated components and on the probability for a component to be taken from mutant vector (mutation probability) is analyzed for several variants of weighting factor and crossover factor, including classical binomial and exponential strategies. For each weighting and crossover variant the relationship between the crossover and mutation probability is identified and its impact on the choice and adaptation of control parameters is analyzed numerically and graphically. Ten different strategies (variations) of DE with penalty function approach are analyzed with various population sizes, crossover and weighting factors and applied to the problem of minimizing the cost of the active parts of the power objects. Constraints resulting from international specifications are taken into account. The Objective functions that are optimized are minimizations dependent on multiple input variables. All constraints are normalized and modeled as inequalities.

Keywords: Optimization methods · Differential evolution · Weighting factor · Crossover constant · Binomial crossover · Exponential crossover · Distribution transformer.

1 Introduction

The Evolutionary Algorithms(EAs) have been successfully applied to solve optimization problems [1], [2], [8]. However, in their original versions, EAs lack a mechanism to deal with the constraints of a problem. The most popular approach is the use of penalty functions [1]. The aim of penalty functions is to decrease the fitness value of those infeasible individuals (which do not satisfy the constraints of the problem). DE shares similarities with traditional EAs. However it does not use binary encoding as a simple genetic algorithm [6], [7] and it does not use a probability density function to self-adapt its parameters as an Evolution Strategy [7]. Instead, DE performs mutation based on the distribution of the solutions in the current population. In this way, search directions and possible step-sizes depend on the location of the individuals selected to calculate the mutation values. The aim of this paper is to analyze both from a numerical and graphical point of view the influence the weighting factor and crossover variant have on the behavior of DE.

A simple additive penalty function approach is used in order to convert the constrained problem into an unconstrained problem. Due to this conversion, the solution falling outside the feasible region is penalized and the solving process is guided to fall into the feasible solution space after a few generations. The method of penalty function approach is very sensitive when the penalty parameters are large. Penalty functions tend to be very sensitive near the boundary of the feasible domain and that result in a local optimal solution or an infeasible solution. It is always necessary to have careful selection of the penalty parameters for the proper convergence to a feasible optimal solution.

2 Related work

Other authors have proposed different approaches to solve constrained optimization problems with DE-based algorithms. A feasible region shrinking mechanism was proposed by Storn [6], [7]. B. V. Babu and M. Mathew Leenus Jehan in [8] have applied Differential Evolution with a Penalty Function Method and Weighting Factor Method for finding a Pareto optimum set for the different problems. DE is found to be robust and faster in optimization. DE managed to give the exact optimum value within less generations compared to a simple Genetic Algorithm.

Mezura-Montes and Coello Coello in [10] present a Differential-Evolution based approach to solve constrained optimization problems. Three selection criteria based on feasibility are used to deal with the constraints of the problem and also a diversity mechanism is added to maintain infeasible solutions located in promising areas of the search space. The conventional DE algorithm highly depends on the chosen trial vector generation strategy and associated parameter values used. DE researchers have suggested many empirical guides for choosing trial vector generation.

Storn and Price [7] suggested that a reasonable value for NP should be between 5D and 10D, and a good initial choice of F was 0.5. The effective range of F values was suggested between 0.4 and 1. The first reasonable attempt of choosing CR value can be 0.1. However, because the large CR value can speed up convergence, the value of 0.9 for CR may also be a good initial choice if the problem is near unimodal or fast convergence is desired. Moreover, if the population converges prematurely, either F or NP can be increased.

Recently, Rönkkönen in [12] suggested using F values between [0.4,0.95] with 0.9 being a good initial choice. The CR values should lie in [0,0.2] when the function is separable while in [0.9,1] when the function's parameters are dependent. However, when solving a real engineering problem, the characteristics of the problem are usually unknown. Hence, it is difficult to choose the appropriate CR value in advance.

Zaharie proposed a parameter adaptation for DE (ADE) based on the idea of controlling the population diversity, and created a multi-population approach [13]. Following the same ideas, Zaharie and Petcu designed an adaptive Pareto DE algorithm for multi-objective optimization and analyzed its parallel version.

The researchers have developed some techniques to avoid manual tuning of the control parameters. For example, linearly reduced the scaling factor F with increasing generation count from a maximum to a minimum value, or randomly varied F in the range (0.5,1). They also have employed a uniform distribution between 0.5 and 1.5 (with a mean value of 1) to obtain a new hybrid DE variant.

The researchers have developed some techniques to avoid manual tuning of the control parameters. For example, Das et al. [18] linearly reduced the scaling factor F with increasing generation count from a maximum to a minimum value, or randomly varied F in the range (0.5,1). They also have employed a uniform distribution between 0.5 and 1.5 (with a mean value of 1) to obtain a new hybrid DE variant [19].

3 The Differential Evolution (DE) algorithm

Differential Evolution (DE) algorithm is a population-based stochastic method for global optimization developed by Rainer Storn and Kenneth Price [6], [7] for optimization problems over continuous domains.

The original version of DE with constituents can be defined as follows ([8], [11]):

- *The population*

The current population, denoted by $P_{x,g}$, is composed ofindividual encoded candidates $\boldsymbol{x}_{i,g}$.

$$\begin{aligned} P_{x,g} &= \left(\boldsymbol{x}_{i,g}\right), \quad i = 0,1,\ldots\ldots,NP, \quad g = 0,1,\ldots\ldots,g_{max} \\ \boldsymbol{x}_{i,g} &= \left(x_{j,i,g}\right), \; j = 0,1,\ldots\ldots,D-1. \end{aligned} \tag{1}$$

where NP is the number of population vectors, g defines the generation counter, and D the number of parameters. The generating a new candidate in the differential algorithm by adding and subtracting vectors is shown in Fig. 1.

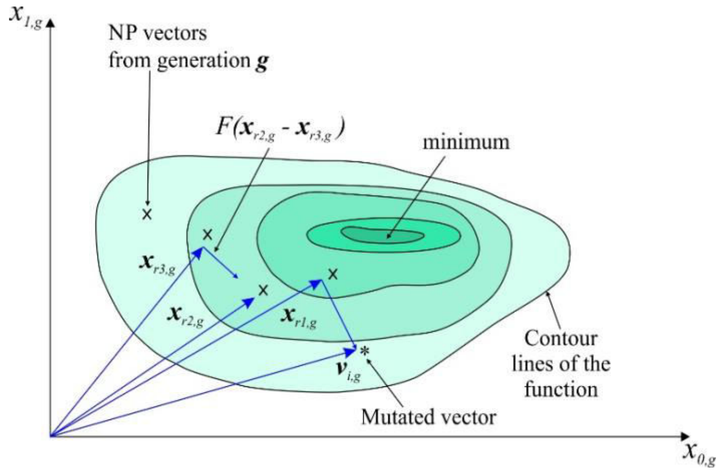


Fig. 1. Generate a new candidate in the differential evolution by adding and subtracting vectors

- *The initialization of the population through*

$$x_{j,i,0} = rand_j\left[0,1\right)\cdot\left(b_{j,max} - b_{j,min}\right) + b_{j,min}\,. \tag{2}$$

The D-dimensional initialization vectors, $\boldsymbol{b}_{min}$ and $\boldsymbol{b}_{max}$ indicate the lower and upper bounds of the parameter vectors $\boldsymbol{x}_{i,j}$. The random number generator, $rand_j[0,1)$, re-turns a uniformly distributed random number from within the range [0,1), i.e., $0 \leq rand_j[0,1) < 1$. Indication that a new random value is generated for each parameter is denoted by the subscript j, Fig. 2.

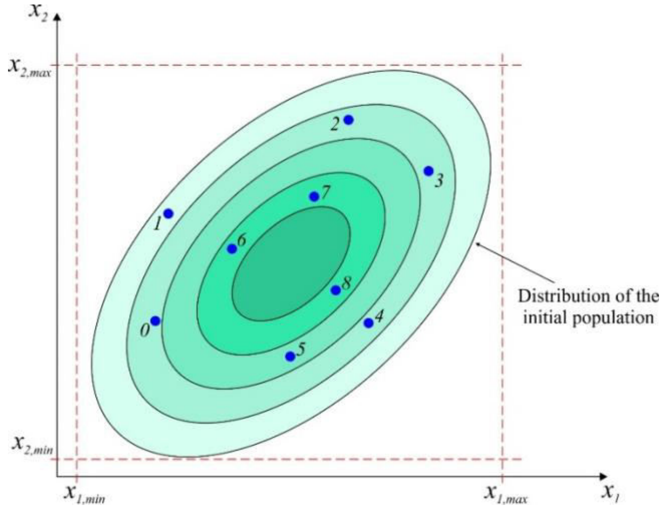
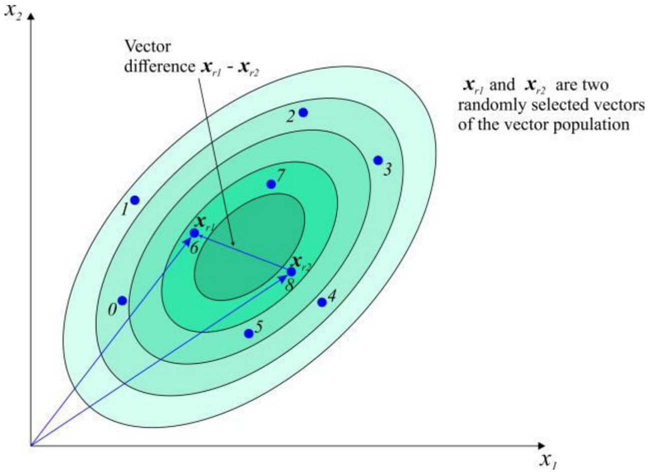


Fig. 2. Initialization of the population in differential evolution

- *The pertubation of a base vector* $\boldsymbol{y}_{i,g}$ *by using a difference vector mutation*

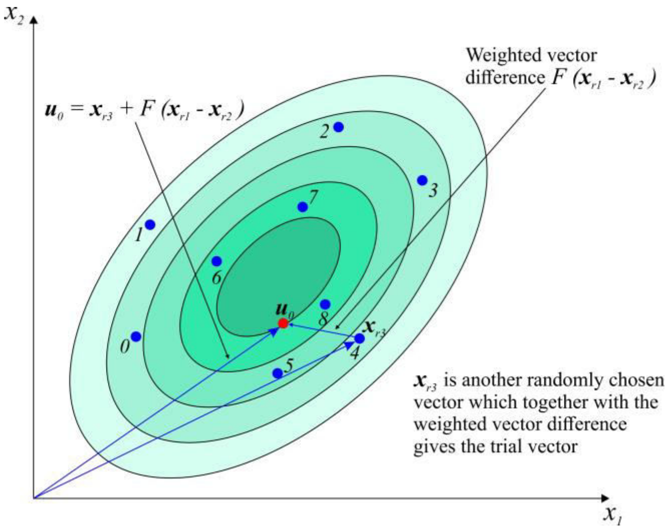
$$\boldsymbol{v}_{i,g} = \boldsymbol{y}_{i,g} + F \cdot \left(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g} \right). \tag{3}$$

to generate mutation vector $\boldsymbol{v}_{i,g}$. The difference vector indices, r_1 and r_2, are randomly selected once per base vector Fig.3, Fig. 4.



$\boldsymbol{x}_{r1}$ and $\boldsymbol{x}_{r2}$ are two randomly selected vectors of the vector population

Fig. 3. Perturbation of two randomly selected vectors



$\boldsymbol{x}_{r3}$ is another randomly chosen vector which together with the weighted vector difference gives the trial vector

Fig. 4. Differential evolution

- *Discrete recombination(crossover)*

The classic variant of diversity enhancement is crossover which mixes parameters of the mutation vector $\boldsymbol{v}_{i,g}$ and the so-called ***target vector*** $\boldsymbol{x}_{i,g}$ in order to generate the ***trial vector*** $\boldsymbol{u}_{i,g}$. The most common form of crossover is uniform and is defined as

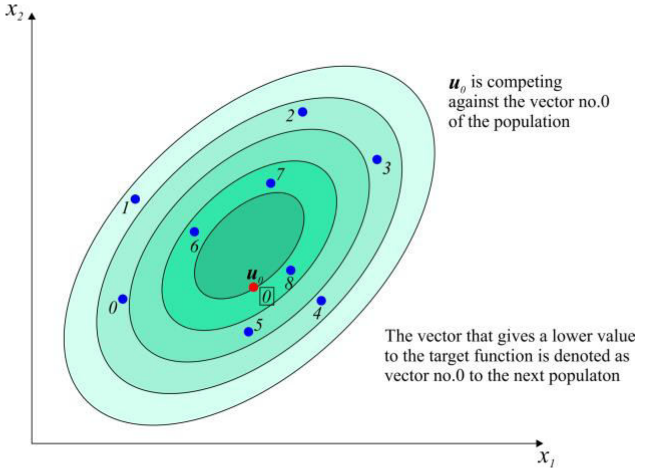
$$\boldsymbol{u}_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g} \ \ if\left(rand_j\left[0,1\right) \le CR\right) \\ x_{j,i,g} \ \ otherwise \end{cases} \tag{4}$$

In order to prevent the case $\boldsymbol{u}_{i,g} = \boldsymbol{x}_{i,g}$ at least one component is taken from the mutation vector $\boldsymbol{v}_{i,g}$, a detail that is not expressed in (4).

- *Selection*

DE uses simple one-to-one survivor selection where the trial vector $\boldsymbol{u}_{i,g}$ competes against the target vector $\boldsymbol{x}_{i,g}$. The vector with the lowest objective function value survives into the next generation $g + 1$, Fig. 5.

$$\boldsymbol{x}_{i,g+1} = \begin{cases} \boldsymbol{u}_{i,g} \ \ if\ f\left(\boldsymbol{u}_{i,g}\right) \le f\left(\boldsymbol{x}_{i,g}\right) \\ \boldsymbol{x}_{i,g} \ \ otherwise. \end{cases} \tag{5}$$



$\boldsymbol{u}_0$ is competing against the vector no.0 of the population

The vector that gives a lower value to the target function is denoted as vector no.0 to the next populaton

Fig. 5. Selection. The vector $\boldsymbol{u}$ has the lower objective function value than the target vector **0** so it moves forward in the next generation

When is the global optimum achieved? It depends on the target function. If the function is a multi-objectives function, the targets may conflict with each other. Satisfying one goal often leaves another unsatisfied. As a user regularly defines the input, the user can also limit the number of iterations to the algorithm. This is a trial and error approach, requiring a sufficient number of repetitions to ensure that the best results are returned. Another method of termination is when the goal is fulfilled. In some target functions, the minimum value can be known. For example, any minimum value may already be known for a function, which means that any subsequent value found under this value is not taken into account. This is also a method used when working with functions where the minimum is known. If it is known that the value of the target function in the farthest vector is within a certain tolerance of the global minimum, the termination is fulfilled.

Human monitoring can also be determined when optimization is complete. The return data provided by the target function can determine that no further optimization is possible.

Along with the DE algorithm came a notation to classify the various DE-variants. The notation is defined by DE/*x/y/z* where *x* denotes the base vector, *y* denotes the number of difference vectors used, and z representing the crossover method. Price and Storn [6] gave the working principle of DE with single strategy [7]. They suggested ten different strategies for DE. The following are the ten different DE working strategies: 1. DE/rand/1/bin, 2. DE/best/1/bin, 3. DE/rand/2/bin, 4. DE/best/2/bin 5. DE/rand-to-best/1/bin 6. DE/rand/1/exp, 7. DE/best/1/exp 8. DE/rand/2/exp, 9. DE/best/2/exp, 10. DE/rand-to-best/1/exp.

A strategy that works out to be the best for a given problem may not work well when applied to a different problem. Also, the strategy and the key parameters to be adopted for a problem are to be determined by trial and error. However, strategy-1 (DE/rand/1/bin) appears to be the most successful and the most widely used strategy. More details regarding DE are available in [7], [8], [9] and [11].

4 Mathematical modeling and optimization of three phase distribution transformer

In the mathematical notation the optimization problem can generally be represented as a pair (S, f), where $S \subseteq R^n$ is a bounded set on R^n and $f: S \rightarrow R$ is an n-dimensional real-valued function. The problem is to find a point $\boldsymbol{x}_{min} \in S$ such that $f(\boldsymbol{x}_{min})$ is a global minimum on S. More specifically, it is required to find an $\boldsymbol{x}_{min} \in S$ such that

$$\begin{aligned} & min\left(f\left(\boldsymbol{x}\right)\right), \quad \boldsymbol{x}=\left(x_1,....,x_n\right)^T \in S \subseteq R^n \\ & \text{with constraints} \\ & \begin{cases} g_j\left(\boldsymbol{x}\right) \leq 0 & j=1,....,q \\ h_j\left(\boldsymbol{x}\right)=0 & j=q+1,....,m \\ x_{i,\min} \leq x_i \leq x_{i,\max} \end{cases} \end{aligned} \tag{6}$$

where $g_i(\boldsymbol{x})$ and $h_j(\boldsymbol{x})$ are the restriction constraints, which can be represented mathematically as equations and/or inequations.

Under these definitions, a DE algorithm in conjunction with the penalty function approach is focused on the minimization of the cost of the transformer unit:

$$\min_{x} \sum_{j=1}^{2} c_j \cdot f_j(\boldsymbol{x}) \tag{7}$$

where c_1 is the winding unit cost (€/kg), f_1 is the winding weight (kg), c_2 is the magnetic material unit cost (€/kg), f_2 is the magnetic material weight (kg), and **x** is the vector of the five design variables, namely the width winding (X_3), the diameter of core leg (X_2), the winding height (X_5), the current density of winding (X_4) and the magnetic flux density (X_1) respectively for each object.

The minimization of the cost of the high voltage single phase transformer is subject to the constraints:

$$S - S_N \le 0;\ P_{CU} - P_{CUN} \le 0;\ P_{FE} - P_{FEN} \le 0;\ U_K - U_{KN} \le 0 \tag{8}$$

where: S is designed transformer rating (kVA), S_N is transformer nominal rating (kVA), P_{FE} is designed no-load losses (W), P_{CU} is designed load losses (W), U_K is designed short-circuit impedance (%), P_{FEN} is guaranteed no-load losses (W), P_{CUN} is guaranteed load losses (W) and U_{KN} is guaranteed short-circuit impedance (%).

Accordingly, the objective function for the model is:

$$\begin{aligned} f(x_2,x_3,x_5) &= \left(3.9655 \cdot 10^4 \cdot x_5 + 2.40546 \cdot 10^5 \cdot x_3 + 2.987 \cdot 10^3\right) \cdot x_2^2 + \\ &1.8924 \cdot x_2^3 + \left(6.96522 \cdot 10^5 \cdot x_2 + 1.42442 \cdot 10^6 \cdot x_3 + 1.3478 \cdot 10^4\right) \cdot x_3 \cdot x_5 \end{aligned} \tag{9}$$

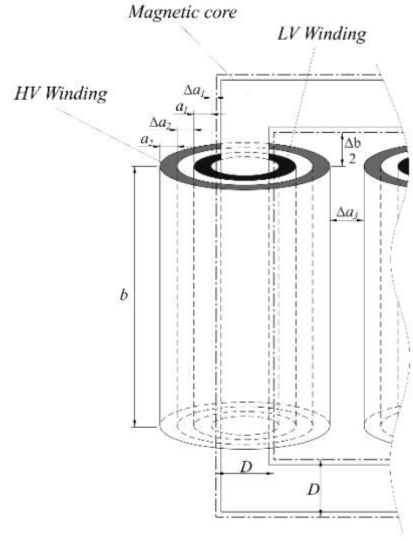


Fig. 6. Active part of three phase distribution transformer – main dimensions

The inequality constraints should be modified to the less or equal format, g($\boldsymbol{x}$) ≤ 0. If the problem is an unconstrained optimization problem, the user need not enter anything in the space specified for the constraints coding. The constraints of the analyzed mathematical model are entered as follows: Constraint 10 match to transformer nominal rating, Constraint 11 match to guaranteed load losses, Constraint 12 match to guaranteed no-load losses and Constraint 13 guaranteed short-circuit impedance. Constants in front of decision variables have been taken from the *Fig.1* and reference [9], [14].

$$317.82 \cdot x_1 \cdot x_2^2 \cdot x_3 \cdot x_4 \cdot x_5 \cdot 10^6 - 50 \cdot 10^3 \leq 0 \tag{10}$$

$$\begin{gathered}\left(3.638 \cdot 10^{-7} \cdot x_2 + 8.113 \cdot 10^{-7} \cdot x_3 + 7.51 \cdot 10^{-9}\right) \cdot \\ \cdot x_3 \cdot x_4^2 \cdot x_5 - 1050 \leq 0\end{gathered} \tag{11}$$

$$\begin{gathered}\left(-0.4237 \cdot x_1^2 + 1.2712 \cdot x_1 - 0.0241\right) \cdot \\ \left(\left(3.9655 \cdot 10^4 \cdot x_5 + 2.405 \cdot 10^5 \cdot x_3 + 2.987 \cdot 10^3\right) \cdot x_2^2 + 1.892 \cdot x_2^3\right) \cdot 0.4 - 190 \leq 0\end{gathered} \tag{12}$$

$$\begin{gathered}\left(0.008 \cdot x_2 + 0.0186 \cdot x_2 \cdot x_3 + 0.032 \cdot x_3 + 1.7744 \cdot x_3^2 + 1.6 \cdot 10^{-4}\right) \cdot 317.82 \cdot \\ 0.0186 \cdot x_3 \cdot x_4 / x_1 \cdot x_2^2 - 4.1 \leq 0\end{gathered} \tag{13}$$

These values are multiplied by a penalty co-efficient, which is then added to the objective function to continue the process of optimization. This process is often termed as penalty function approach. From Table 1 for outcomes:
Best Strategy is Strategy DE/rand/1/bin
Minimum constraint violation(CV): 0.00
Minimum objective value with min CV:497.4702
Minimum time taken: 78

Table 1. Results of DE with all ten strategies

Str. No.	Strategy	NP	CR	F	Optimal Value	CV	Time (ms)*
1	**DE/rand/1/bin**	**1000**	**0.9**	**0.5**	**497.4702**	**0.00**	**78**
2	DE/best/1/bin	1000	0.9	0.5	497.3557	0.00	94
3	DE/best/2/bin	1000	0.9	0.5	497.3257	0.00	140
4	DE/rand/2/bin	1000	0.9	0.5	497.4702	0.00	79
5	DE/rand-to-best/1/bin	1000	0.9	0.5	497.3322	0.00	78
6	DE/rand/1/exp	1000	0.9	0.5	497.4702	0.00	78
7	DE/best/1/exp	1000	0.9	0.5	497.3193	0.00	94
8	DE/best/2/exp	1000	0.9	0.5	497.3623	0.00	94
9	DE/rand/2/exp	1000	0.9	0.5	497.4702	0.00	78
10	DE/rand-to-best/1/exp	1000	0.9	0.5	497.3331	0.00	78

Table 2. The optimal value of decision variables

Parameter	Value
X_1	1.713658
X_2	0.100009
X_3	0.015001
X_4	2.741579
X_5	0.210037

5 Results and discussions

The above mathematical model is solved using Differential Evolution with penalty function approach. Total number of constraints and bounds in the model are 4 and 10 respectively. The model is run for different combinations of ten DE strategies, NP (100 to 850 with an increment of 50), CR (0.2 to 0.90 with an increment of 0.1) and F (0.2 to

0.9 with an increment of 0.1) to determine the optimum value of the objective function. The best combination of NP, CR and F which will yield minimum value of objective function from all the different combinations is chosen for each strategy. These are presented in Table1. In addition, comparison of CPU time in seconds for each strategy is also performed which is based on PC with PV 2.67GHz/6GB RAM/300GB HDD(*). The following observations are made from the analysis of results.

- It is evident from Table1 that strategy 1, DE/rand/1/bin with NP = 1000, CR = 0.90 and F = 0.50 is the best strategy as it produces minimum CPU time with no constraint violation.
- On the other hand all strategies are yielding approximately same optimal values of objective function with minor difference even small differences in CPU time. Reaching the same optimal values from number of strategies maybe due to the sufficient resources (lower and upper bounds) that are available to satisfy demands.
- It is observed from Fig.7 that the strategies with binomial crossover are closer to the optimum.
- Fig.8, Fig.9, Fig.10 and Fig.11 shows the effect of varying weighting factor F (0.2 to 0.90 amounting to 8 levels), NP values (100 to 800 amounting to 8 levels) for 4 different strategies, namely, DE/rand/1/bin, DE/best/1/bin, DE/rand/1/exp and DE/best/1/exp (keeping CR=0.90) on value of objective function. The minimum dispersing of value of objective function is for weighting factor F from 0.40 to 0.6. It is inferred from *Figs*.8 to 11 that lower value of F gives a higher chance of convergence to the optimum and there is a gradual decrease in the optimal value as F increases.
- It is also observed that values of objective function are converging to optimum value with CR= 0.90 as high values of CR results in a rotationally invariant sampling of the search space and helps in a faster and/or more robust convergence [8].
- Fig.14 shows the variation of population size NP for 4 different strategies namely, DE/rand/1/bin, DE/best/1/bin, DE/rand/1/exp and DE/best/1/exp for a fixed value CR=0.9 and F=0.4. It is observed that the variation of NP does not follow a regular pattern and it needs a trail and error procedure to determine the optimum NP value.

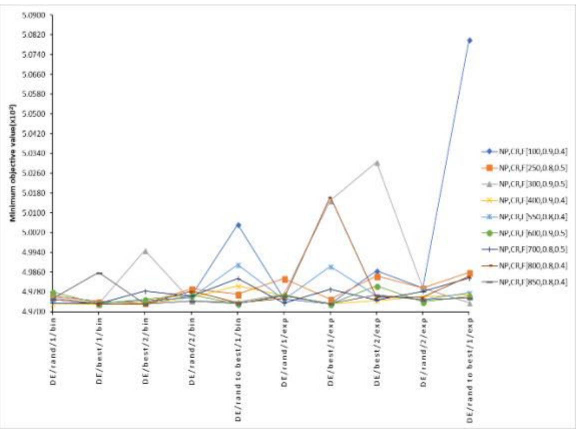


Fig. 7. Strategy variation for a sample set of parameters NP, CR&F

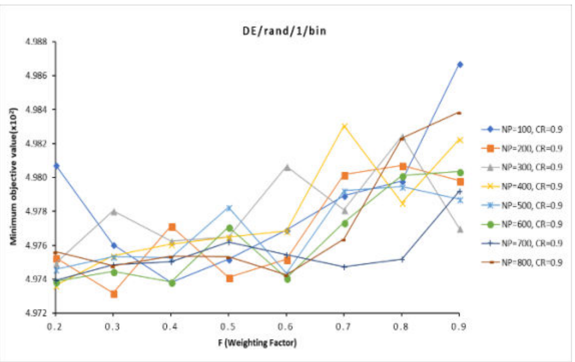


Fig. 8. Variation of weighting factor F for different NP values keeping CR=0.90 for strategy DE/rand/1/bin

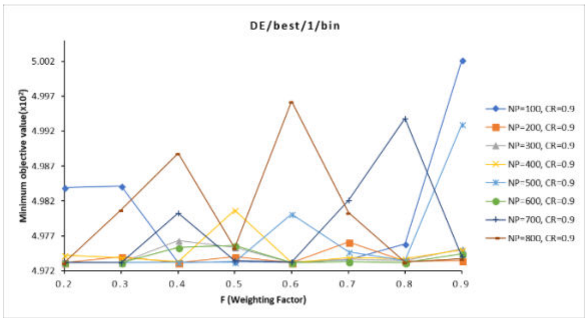


Fig. 9. Variation of weighting factor F for different NP values keeping CR=0.90 for strategy DE/best/1/bin

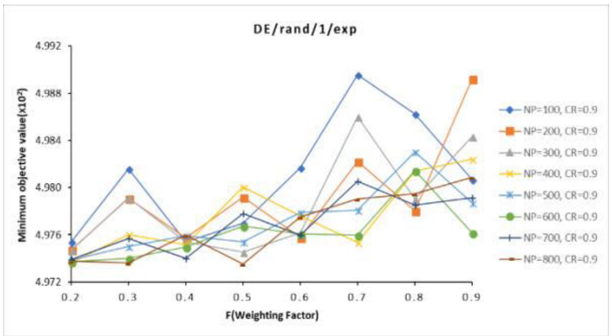


Fig. 10. Variation of weighting factor F for different NP values keeping CR=0.90 for strategy DE/rand/1/exp

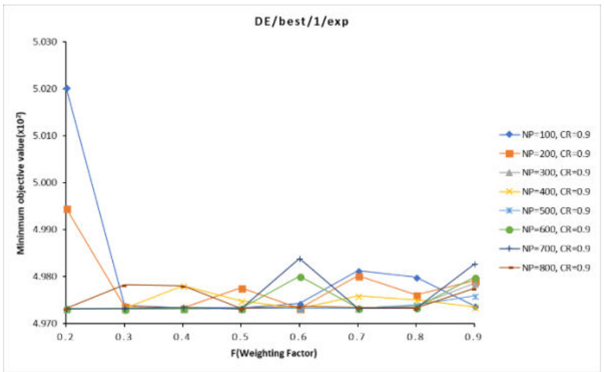


Fig. 11. Variation of weighting factor F for different NP values keeping CR=0.90 for strategy DE/best/1/exp

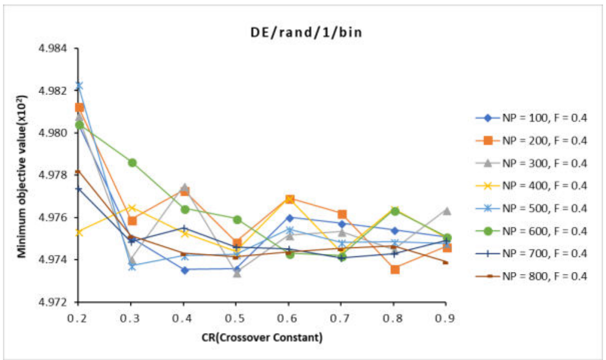


Fig. 12. Variation of crossover constant CR for different NP values keeping F=0.4 for strategy DE/rand/1/bin

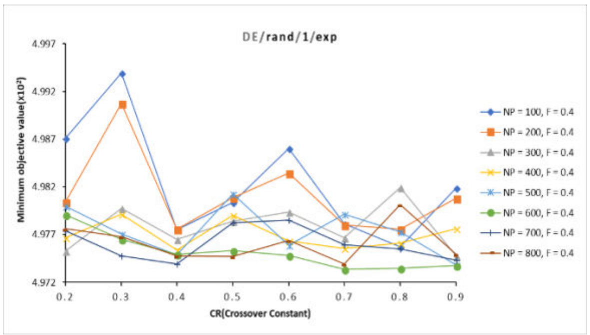


Fig. 13. Variation of crossover constant CR for different NP values keeping F=0.4 for strategy DE/rand/1/exp

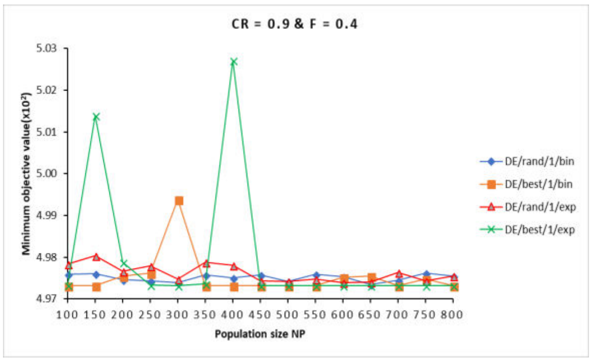


Fig. 14. Variation of Population size NP for different strategies at F= 0.4 and CR= 0.9

6 Conclusion

This paper presents an efficient implementation of the Differential Evolution algorithm with a penalty function approach, applied to three phase distribution transformer unit. Our penalty function approach integrates established techniques in existing EA's in a single unique algorithm.

Effect of Population size(NP), Crossover constant(CR), Weighting factor (F) and ten different strategies(variations) of DE on the values of objective function are studied. Results of DE are also compared for all ten strategies. It is concluded that DE/rand/1/bin is the best strategy for the optimization of the cost material of the three phase distribution transformers active part. The best approach is with crossover constant 0.9 and weighting factor 0.4. The present study can be extended to similar situations with suitable modifications (different power objects).

References

1. R. Salkoski, “Heuristic algorithm for multicriterial optimization of power objects”, PhD Thesis, University” SS. Cyril and Methodius” FEIT, Skopje, R. Macedonia (2018)
2. A. Vasan, K. Srinivasa R.: “Optimal Reservoir Operation Using Differential Evolution” Bisalpur drinking water cum irrigation project report, Government of Rajasthan, India. (2004)
3. A. Zamuda, J. Brest, B. Bošković, V. Žumer, “Differential Evolution with Self-adaptation and Local Search for Constrained Multi-objective Optimization,” IEEE Congress on Evolutionary Computation (CEC), pp. 195-202 (2009)
4. DE Homepage, http://www.icsi.berkeley.edu/~storn/code.html
5. Onwubolu, G. C., and Babu, B. V.: New Optimization Techniques in Engineering, Springer-Verlag, Germany (2004)
6. P.V. Kenneth., S.M. Rainer.: Differential evolution - A simple evolution strategy for fast optimization. Dr. Dobb's Journal, 22, 18-24 and 78 (1997)
7. P.V. Kenneth., S.M. Rainer., L.A. Jouni.: Differential evolution: A practical approach to global optimization. Springer-Verlag, Berlin, Heidelberg (2005)
8. B.V. Babu, M. Mathew, L. Jehan.: Differential Evolution for Multi-Objective Optimization. Chemical Enginering Department B.I.T.S. Pilani, India (2005)
9. R. Salkoski.: Selection of an optimal variant of 3-phase transformers with round and rectangular section of the magnetic core from aspect of minimum production costs. Master Thesis, Electrotechnical University in Skopje (2000)
10. Mezura and Montes.: E. Laboratorio NI Avanzada, Rébsamen 80, Centro, Xalapa, Veracruz 91090, Mexico, Velazquez-Reyes, J., Coello Coello, C.A.: Modified Differential Evolution for Constrained Optimization, pp 25 – 32, Conference Publications, Evolutionary Computation, CEC 2006 (2006)
11. U.K. Chakraborty (Ed.).: Advances in Differential Evolution, Mathematics & Computer Science Department, University of Missouri, St. Louis, USA, Springer-Verlag Berlin Heidelberg (2008)
12. J. Rönkkönen, S., Kukkonen, K. V. Price.: Real-parameter optimization with differential evolution. Proc. IEEE Congr. Evolut. Comput., Sep. 2005, pp. 506–513, Edinburgh, Scotland (2005)
13. D. Zaharie.: Influence of crossover on the behavior of differential evolution algorithms. Applied Soft Computing 9, 1126-1138 (2009)
14. R. Salkoski, I. Chorbev.: Design optimization of distribution transformers based on Differential Evolution Algorithms, ICT Innovations2012 Web Proceedings ISSN 1857-7288, page 35-44 (2012)
15. M. Weber, F. Neri, V. Tirronen.:” A study on scale factor in distributed differential evolution, Information Sciences 181(12), 2488-2511 (2011)