International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 8 (2016) pp. 301-311

Application of Aesthetic Differential Evolution in Identification of Noisy Sources in a Multi Noise Plant

Tarun Kumar Sharma[1] and Millie Pant[2]

[1] Amity University Rajasthan,
NH 11 C, Kant Kalwar 303002, Jaipur, Rajasthan, India
taruniitr1@gmail.com

[2] Department of Applied Science and Engineering, Saharanpur Campus, IIT Roorkee,
Paper Mill Road, 247001, Saharanpur, India
millidma@gmail.com

Abstract*: The applications of Differential Evolution (DE) and its variants narrate its success. DE is simple, efficient and powerful stochastic optimization algorithm to solve wide range of optimization problems. But sometimes it gets stuck into local optima that results in slow convergence. To overcome this issue, we have proposed a modified variant of DE called aesthetic DE algorithm (ADEA). In this proposed variant the mutation phase is modified to generate the new positions using the concept of reflection. The position of global best individual is reflected i.e. mirror image to get new positions (solutions). This concept provides perturbation that in later stage helps in getting optimal positions. The proposed variant is tested on a set of 13 benchmark functions consulted from literature. The simulated results are then compared with basic DE and state-of-art algorithms. Non-parametric statistical analysis is performed for the result comparisons. Further ADEA is investigated and also compared with DE and on a real time problem of identification of noisy sources in a multi noise plant. The experimental results show the efficacy of the proposal.

Keywords: Differential Evolution, DE, Multi Noise Plants, Mutation, Noise Identification, Global Optimization

I. Introduction

Differential Evolution (DE) introduced in 1995 by Storn & Price [1], is a simple, efficient and powerful stochastic optimization algorithm to solve wide range of optimization problems. The applications of Differential Evolution (DE) and its variants in versatile domains narrate its success. The efficiency of DE over its counterpart algorithm such as simulated annealing (SA), evolutionary programming (EP), controlled random search (CRS), genetic algorithm (GA) and particle swarm optimization (PSO) can be consulted from [2] – [4].

Despite many positive features, DE also has certain drawbacks associated with it. It is susceptible to problems like slow and/ or premature convergence, is sometimes unable to locate global optima or gets stuck in local optima. Also, like most of the other population based evolutionary algorithm (EA) / Nature Inspired Algorithm (NIA), the performance of DE deteriorates with the increase in the size of the problem. These drawbacks of DE become more persistent in case of multimodal or noisy functions.

Scaling factor (F) and Crossover rate (Cr) are the two control factors apart from population size in DE. Parameter tuning plays a vital role in the success of algorithm. Inappropriate parameter tuning or selection of strategy sometimes may limit to convergence speed or may trigger premature convergence escaping local minima. The details are reasonably demonstrated with the values of F and Cr in [3] [5] and [4] [6].

As there is no free lunch algorithm which can be equally applied to solve almost every kind of problem. The same is also true for DE. DE looses its efficiency in terms of slow convergence while handling noisy problems. This study presents aesthetic differential evolution algorithm (ADEA) to enhance the efficiency and performance of the conventional DE to solve optimization problems. In the proposed variant of DE called ADEA, the new positions (decorative positions) are produced using the concept of mirror images. The concept is inspired from interior search algorithm.

The mirror is placed near the most beautiful (global best) individual to accentuate their attractiveness. This idea facilitates in providing the displacement of optimal solutions (based on fitness value). The individual solution in the population is updated only when a better fitness value is achieved.

This dynamic behavior assists in dealing with complex optimization problems. This mechanism helps in following optimal solutions.

Here we would like to add that the preliminary version of this study has been presented in the 7th World Congress on Nature and Biologically Inspired Computing (NaBIC '15) [7]. In that study the proposed variant was applied to solve ten Computationally Expensive Optimization (CEO) problems. In this study, we have extended ADEA implementation in solving thirteen benchmark problems [4] and a real time application of detecting the locations of noisy equipments in multi noise plant [8].

In general life the sound that creates or causes disturbance is termed as Noise. In any factory or industry the high level noise can be very harmful to the employees. As investigated by Occupational Safety and Health Act (OSHA) of 1970, the high level noise not only causes physiological ailments in employees but also causes harmful environment in the neighborhood. Therefore it becomes essential to control the noise levels in any manufacturing plant or industry. This can be achieved by optimal allocation of noise equipments which is quite not easy to recognize the exact location. So, ADEA is applied to identify the locations. In the present study two cases are considered (1) plant with one equipment and (2) plant with two equipments. The detail of the problem is discussed in Section 5.

The outline of the paper is as follows: brief overview of DE is presented in Section 2. ADEA, the proposed variant of DE is detailed in Section 3. Section 4 demonstrates the benchmark problems parameter settings and experimental results. Section 5 presents the mathematical formulation for noise recognition in a multi noise plant. Simulated results are presented in Section 6. In Section 7, the conclusions drawn from the study are summarized.

II. Differential Evolution (DE) Algorithm: An Overview

Since last decades DE has shown a remarkable success when applied to solve real world optimization problems. However, like other its counterpart, DE sometimes gets struck in local optima that lead to slow convergence. In order to enhance its ability and performance number of DE variant have been proposed. In the next paragraph some of the recent modifications in the structure of basic DE are presented.

A self adaptive variant of DE is given by Ghosh et al. [9] in 2011. In that study authors introduced the controlling of scaling (F) and crossover (Cr) parameters at run time. In the same year Zou et al. [10] proposed a modified variant of DE where these parameters were dynamically adjusted with successive iterations. Authors also investigated its performance on assignment problem.

Wang et al. [11] in 2012, presented a variant of DE in which the two antagonist (exploration and exploitation) are balanced while maintain the diversity of solutions. This variant is applied to solve binary coded optimizations problems. To show its performance a problem of multidimensional knapsack is solved. In the same year Mohamed and Sabry [12] presented a new directed mutation rule. The variant is applied to solve constrained optimization problems.

Zou et al. [13] in 2013, proposed modified DE that employs Gaussian and uniform distribution to adjust the values of controlling factors (F and Cr). In the same year, Wu et al. [14] a presented a variant of DE to improve global searching capability. In this self adaptive DE, the population is dynamically divided and control parameters (F & Cr) are also dynamically adjusted on run time.

Ali et al. [15] in 2014 applied basic DE in enhancing the quality of watermarked image. Coelho et al. [16] proposed self adaptive DE that uses gamma distribution, Gaussian probability and chaotic sequence.

Mohamed [17] in 2015 presented a new mutation rule (triangular mutation) in basic DE. This rule is based on convex combination vector of the triplet. A variant of DE that takes its inspiration from Shuffled frog leap algorithm in order to increase the convergence of basic DE is proposed by Xiang et al. [18]. A surrogate model of DE is introduced by Mallipeddi and Lee [19] to tune the control parameters (F & Cr).

Zamuda and Brest in 2015 [20] proposed adaption and self adaption mechanism in DE to generate new control parameter values. Cai and Wang also in 2015 [21] embedded linkage learning technique in crossover phase to alleviate the drawback of linkage-blind in DE. Poonia et al. [7], presented a modified variant of DE, called ADEA in which the mutation operator is modified in order to improve the convergence rate. In 2016, Wang and Tang [22] proposed adaptive multi-population DE, where the size of the subpopulation is adjusted based on search criterion. This variant is applied in data mining process. Basu [23] proposed quasi-oppositional based learning DE to optimize reactive power dispatch problem. The concept is applied in both population initializations as well as for generation jumping. Yi et al. [24] introduced a novel variant of DE. They embedded *p*best roulette wheel selection and retention mechanism to improvise the efficiency of DE.

For more readings about the recent enhancements and applications of DE, readers may consult [25].

Like other evolutionary algorithm, DE also starts with a population of NP candidate solution to explore the search space. After initialize the population, DE use repeated cycles of evolutionary operation such as mutation, crossover and selection to guides the population towards the vicinity of the global optimum. At the end of each generation, the algorithm aims to generate the new population by replacing points of current population according as their quality where quality of each vector is evaluated using a fitness function of the problem. Throughout the present work we have followed DE/rand/1/bin strategy which is a most common mutation strategy for DE and referred as the basic version of DE. The basic structure of DE is as follows:

Basic DE

a) Initialization

Let $X_i^G = \{x_{1,i}^{\ G}, x_{2,i}^{\ G}, x_{3,i}^{\ G}, ..., x_{D,i}^{\ G}\}$ denotes the i^{th} individual in population PG of size NP at any generation G. (1)

D represents dimension of the problem in search space.

A uniformly distributed population is generated randomly within the range of lower and upper bounds so that the search space can sufficiently cover as much as possible.

Let the lower and upper bounds of search space are $l_{min} = \{l_1, l_2, ..., l_D\}$ and $u_{max} = \{u1, u2, ..., u_D\}$ respectively.

j^{th} component of the i^{th} individual can be initialized according to the following equation:

$$x_{j,i}^0 = l_{\min j} + rand_{i,j}[0,1] \times (u_{\max j} - l_{\min j}) \tag{2}$$

where $rand_{i,j}$ [0, 1] is uniformly distributed random which limits in 0 and 1 range.

b) Mutation

Mutation is the second phase after initialization. In this phase mutant (perturbed) vector ($V_{i,}^{G}$) corresponding to each target vector ($X_{i,}^{G}$) is generated using the following scheme:

$$V_{i,G} = X_{r1,G} + F \times (X_{r2,G} - X_{r3,G}) \quad (3)$$

where r_1, r_2, r_3 are mutually exclusive random numbers choosen from [1, *NP*] and all are different from base vector *i*. *F*, the amplification lies between 0 & 1.

c) *Crossover*

Crossover phase is activated just after the mutation phase that helps in enhancing population diversity. Crossover operator is applied on each pair of target vector $X_i^G = \{x_{1,i}^{\ G}, x_{2,i}^{\ G}, x_{3,i}^{\ G}, ..., x_{D,i}^{\ G}\}$ and its corresponding perturbed vector $V_i^G = \{v_{1,i}^{\ G}, v_{2,i}^{\ G}, v_{3,i}^{\ G}, ..., v_{D,i}^{\ G}\}$ to introduce trial vectors $U_i^G = \{u_{1,i}^{\ G}, u_{2,i}^{\ G}, u_{3,i}^{\ G}, ..., u_{D,i}^{\ G}\}$. The binomial crossover is performed to produce trail vectors.

$$u_{j,i,G} = \begin{cases} v_{j,i,G} & \text{if } rand_{i,j}[0,1] \le Cr \vee j = j_{rand} \\ x_{j,i,G} & otherwise \end{cases} \quad (4)$$

j_{rand} is randomly chosen index from 1 to D. Crossover rate, the other controlling positive parameter of DE (*Cr*) $\in$[0, 1] is set by the user or may be adaptive. If $rand_{i,j}$ [0,1] ≥ *Cr*, then the components come from the target vector; otherwise, it is taken from mutant vector.

d) Selection

The final phase of DE is Selection, where the population for the next generation is selected on greedy mechanism is given as:

$$X_{i,G+1} = \begin{cases} U_{i,G} & \text{if } f(U_{i,G}) \le f(x_{i,G}) \\ X_{i,G} & Otherwise \end{cases} \quad (5)$$

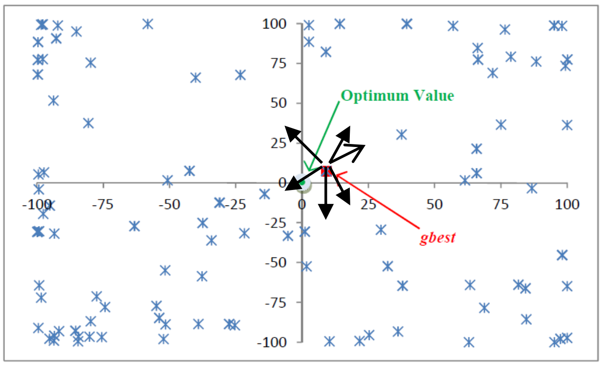
III. Proposed ADEA

ADEA, a new variant of DE is proposed. Enhancing the convergence ability of DE is the basic aim behind this study. In the proposed variant the mutation phase of DE is modified. In this variant new positions are produced using the concept of reflection introduced in Interior Search Algorithm (ISA) [26]. The position of global best individual is reflected i.e. mirror image. This concept provides perturbation that in later stage helps in getting optimal positions (solutions). If better fitness value is achieved then the individual solution in the population is updated. This process assists in getting optimal solutions. ADEA proceed in similar fashion of basic DE. Each solution is estimated using fitness function i.e. objective function. An individual having the minimum value (x_{gbest}) is identified. Then to perform local search a random walk is performed to slightly change the position of x_{gbest} (Fig. 1 & 2(a)) using equation (6). This process enhances exploitation in DE.

$$x_{gbest}^{k} = x_{gbest}^{k-1} + r_{NP} + \lambda \quad (6)$$

where r_{NP} represents uniformly distributed random number and λ is scale factor, which depends on the size of search region and set as:

$$\lambda = 0.01 * (u - l) \quad (7)$$



Random solutions in the search space (**x**)*; Random Walk* (→); *gBest* (■); *Optimum Value* (●)

Figure 1. Randomly generated population with global best (Red mark), optimal (green mark) and blue arrows presents possible random walk

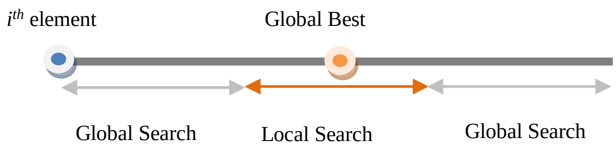
Secondly, a parameter α (ranges between 0 and 1) is introduced to give equal consideration to two antagonists (exploration and exploitation) in the mutation phase. In mutation process of ADEA a mirror is placed randomly between each individual and *gbest*. The idea is demonstrated in Fig 2(b). The position of the mirror for i^{th} individual in k^{th} iteration is given as:

$$x_{m,i,G}^{k} = r_5 x_{i,G}^{k-1} + (1 - r_5) x_{gbest,G}^{k} \quad (8)$$

where r_5 is a random number between 0 and 1.
The reflected image of the individual by mirror is given as:

$$v_{i,G}^{k} = 2x_{m,i,G}^{k} - x_{i,G}^{k-1} \quad (9)$$

The local search is performed when the reflected image (Fig. 2(b)) is near the *gbest* otherwise global search is performed; this helps in exploration and exploitation. If the reflected images fall outside the fixed bounds then they are adjusted using equation (10).



(a)

(b)

Figure 2. (a) Random Walk and (b) Mirror Searching process in ADEA (Adapted from [18]).

If $\alpha \geq 0.5$ (r_6) the mutants are generated using original mutation process given in equation (3) otherwise the mutant vectors generated using mirror images are taken forward for crossover.

The pseudo code of ADEA is detailed in Fig. 3.

Aesthetic Differential Evolution Algorithm

Initialize the population of solutions using equation (2).
while stopping criterion not satisfied
Evaluate the fitness function and find global best (x^k_{gbest}) , where k is iteration.
for i = 1 to NP
  if x_{gbest}
  $x^k_{gbest,G} = x^{k-1}_{gbest,G} + r_{NP} + \lambda$
  if $r_5 \leq \alpha$
    $x^k_{m,i,G} = r_6 x^{k-1}_{i,G} + (1-r_5) x^k_{gbest,G}$
    $v^k_{i,G} = 2x^k_{m,i,G} - x^{k-1}_{i,G}$
  else
    $v^k_{i,G} = x_{r1,G} + F \times (x_{r2,G} - x_{r3,G})$
  end if
  Check the boundaries.
Perform Crossover
Selection
end for
end while

Figure 3. Pseudo-code of ADEA

Method	Best result	Average result
Fuzzy systems	0.24	0.30
Genetic algorithms	0.17	0.28
Neural Networks	0.20	0.27

Table 1. Comparison of results.

IV. Benchmark Problems and Experimental Settings

In order to investigate the efficiency of the proposal a test bed of thirteen single objective benchmark problems is consulted from literature [4]. The details of the experimental settings are as follows:

- Numerical experiments are performed on PC with Celeron (R) Dual Core CPU T3100@1.90GHz and 2 GB RAM under Microsoft Windows XP Professional (2002). The implementation of ADEA is done on Deb C++.
- DE Parameters

The population size is fixed to 100. Scaling (F) and crossover rate (Cr) are fixed as 0.5 and 0.9 respectively and dimension (D) of the problems is taken as 30. Maximum numbers of function evaluations (NFE) are fixed to 105. For all the experiments 25 independent runs are performed. rand (), defined in C++ is used to generate random numbers. Following conditions adapted from [27] are used to handle the boundary constraints:

$$f(x_i) = \begin{cases} r_7 \times l_i + (1-r_7) x_{gbest,i} & if\ x_i < l_i \\ r_8 \times u_i + (1-r_8) x_{gbest,i} & if\ x_i > u_i \end{cases} \quad (10)$$

where r_7 and r_8 are random numbers between 0 and 1. x_{gbest} is global best solution.

- Simulation Strategy

In order to evaluate the performance of the ADEA, the statistical results in terms of best, worst function value, median, mean and standard deviation (Std. Dev.) are evaluated.

Algorithm taken for performance comparisons

—DE/rand/1/bin (F = 0.9, Cr = 0.9) [2]

—DE/rand/1/bin (F = 0.5, Cr = 0.9) [4]

—DE/best/1/bin (F = 0.6, Cr = 0.3) [28]

—jDE [4]

—AuDE [29]

The fact for choosing the above mentioned state-of-art algorithms for result comparison is that these versions keep intact the original mutation operator of basic DE.

V. Theoretical background of Noise Problem

In this study a model to optimize noise control for the m-noise plant is consulted from [8]. Figure 4 presents m-noise plant with m number of equipments. To evaluate the impact of noise in the environment an n-point monitoring system is installed around the boundary line of the plant. Let [$x_{j(act)}$, $y_{j(act)}$, $x_{j(act)}$] and $SWL_{j(act)}$be the actual Cartesian coordinate and sound power level with respect to the j^{th} noise source. The sound pressure level (SPL), as given by distance decay formula, at any point i radiated from the j^{th} noise source with a variable noise reduction is:

$$SPL_i = 10 * \log\{\sum_{j=1}^{M} 10^{SPL_{ij}/10}\}$$

$$SPL_{ij} = SWL_j - \psi_j(R_{ij}, \phi, fr) - 20 * \log(R_{ij}) - 11 \quad (11)$$

$$\psi_j(R_{ij}, \phi, fr) = 7.4(\frac{R_{ij}\, fr^2}{\phi}) * 10$$

where R_{ij} is the distance between i^{th} point to j^{th} noise source. ø and fr is the humidity in the air and sound frequency respectively. (R_{ij}, ϕ, fr) , is the air sound absorption at $20^o\, C$.

International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 8 (2016) pp. 301-311

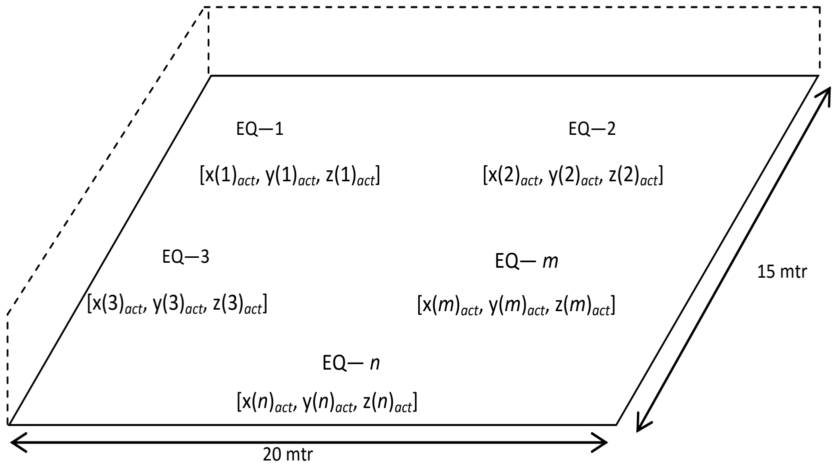


Figure 4. .Location of Equipments (EQ) in Multi Noise Plant

In order to minimize the noise influence in the plant the objective function is given as the total variation of sound level at trail receiving points with respect to the targeted actual noise level. The objective function is defined as:

Minimize

$$f(x_1..x_m; y_1..y_m; z_1..z_m; SPL_{act-1}..SPL_{act-m}; SWL_1..SWL_m = \sum_{i=1}^{N}(SPL_i - SPL_{act-i})^2 \quad (12)$$

Case Study and experimental settings

To verify the efficiency and reliability of the proposed algorithm two cases have been considered. The plant area is considered as 20 (mtr) x 15 (mtr) (*mtr* is meters) on *x* and *y* axis with two kind of *N* – point monitoring system i.e. *N* = 18 and 36. The coordinates of both the monitoring system are illustrated in Table 1 & 2.

The cases are described as follows:

Case – I: Plant with single equipment (*M* = 1)

The corresponding Cartesian coordinates and sound power level of equipment (EQ – 1) is taken as:

—EQ – 1: Cartesian coordinates are (5.0, 7.5, 2.0) and *SWL* = 105.00 db(A).

The boundary conditions are:

0.0 (mtr) ≤ *x* ≤ 20.0 (mtr); 0.0 (mtr) ≤ *y* ≤15.0 (mtr); 0.0 (mtr) ≤ *z* ≤ 3.0 (mtr); 80 0 db(A) ≤ *SWL* ≤120.0 db(A);

Case – II: Plant with single equipment (*M* = 2)

The corresponding Cartesian coordinates and sound power level of equipment (EQ – 1) is taken as:

—EQ – 1: Cartesian coordinates are (5.0, 7.5, 2.0) and SWL_1 = 105.00 db(A).

—EQ – 2: Cartesian coordinates are (18.0, 7.5, 3.0) and SWL_2 = 102.00 db(A).

The boundary conditions for EQ – I & II are:

0.0 (mtr) ≤ x_1 ≤ 20.0 (mtr); 0.0 (mtr) ≤ y_1 ≤15.0 (mtr);0.0 (mtr) ≤ z_1 ≤ 3.0 (mtr);80 0 db(A) ≤ SWL_1 ≤120.0 db(A);

0.0 (mtr) ≤ x_2 ≤ 20.0 (mtr); 0.0 (mtr) ≤ y_2 ≤15.0 (mtr);0.0 (mtr) ≤ z_2 ≤ 3.0 (mtr);80 0 db(A) ≤ SWL_2 ≤120.0 db(A);

M.P*	*x*	*y*	*z*	M.P	*x*	*y*	*z*	M.P	*x*	*y*	*z*
R_1	0.0	0.0	1	R_7	8.0	0.0	1	R_{13}	16.0	0.0	1.0
R_2	0.0	7.5	1	R_8	8.0	7.5	1	R_{14}	16.0	7.5	1.0
R_3	0.0	15.0	1	R_9	8.0	15.0	1	R_{15}	16.0	15.0	1.0
R_4	4.0	0.0	1	R_{10}	12.0	0.0	1	R_{16}	20.0	0.0	1.0
R_5	4.0	7.5	1	R_{11}	12.0	7.5	1	R_{17}	20.0	7.5	1.0
R_6	4.0	15.0	1	R_{12}	12.0	15.0	1	R_{18}	20.0	15.0	1.0

Table 1. Cartesian coordinates of 18 point monitoring system

International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 8 (2016) pp. 301-311

M.P	*x*	*y*	*z*	M.P	*x*	*y*	*z*	M.P	*x*	*y*	*z*
R_1	0.0	0.0	1.0	R_{13}	8.0	0.0	1.0	R_{25}	16.0	0.0	1.0
R_2	0.0	3.0	1.0	R_{14}	8.0	3.0	1.0	R_{26}	16.0	3.0	1.0
R_3	0.0	6.0	1.0	R_{15}	8.0	6.0	1.0	R_{27}	16.0	6.0	1.0
R_4	0.0	9.0	1.0	R_{16}	8.0	9.0	1.0	R_{28}	16.0	9.0	1.0
R_5	0.0	12.0	1.0	R_{17}	8.0	12.0	1.0	R_{29}	16.0	12.0	1.0
R_6	0.0	15.0	1.0	R_{18}	8.0	15.0	1.0	R_{30}	16.0	15.0	1.0
R_7	4.0	0.0	1.0	R_{19}	12.0	0.0	1.0	R_{31}	20.0	0.0	1.0
R_8	4.0	3.0	1.0	R_{20}	12.0	3.0	1.0	R_{32}	20.0	3.0	1.0
R_9	4.0	6.0	1.0	R_{21}	12.0	6.0	1.0	R_{33}	20.0	6.0	1.0
R_{10}	4.0	9.0	1.0	R_{22}	12.0	9.0	1.0	R_{34}	20.0	9.0	1.0
R_{11}	4.0	12.0	1.0	R_{23}	12.0	12.0	1.0	R_{35}	20.0	12.0	1.0
R_{12}	4.0	15.0	1.0	R_{24}	12.0	15.0	1.0	R_{36}	20.0	15.0	1.0

Table 2. Cartesian coordinates of 36 point monitoring system

VI. Result Discussion

A. Benchmark Problems

The compiled results of thirteen benchmark problems in terms of mean and standard deviation (Std.) of ADEA and the state-of-art algorithms taken for comparison are presented in Table 3 and 4.

In Table 3 the results of ADEA are compared with DE having different control parameters and mutation strategy. Table 4 illustrates the result comparison of ADEA with jDE and AuDE. Best results are highlighted (bold) in both the Tables.

It can be clearly observed from the Tables 3 and 4 that ADEA performed consistently well for all the thirteen problems, especially in case of P_2, P_3, P_4, P_5, P_7, P_8 and P_{11} where as in case of P_1, P_6, P_9, P_{10}, P_{12}, P_{13} it can be analyzed that the proposed ADEA present a competitive performance with respect to the other state-of-art algorithms taken for comparison.

Further to test the efficiency of the proposal, non-parametric test suggested in [31] & [32] is also performed. Post-hoc test is performed using Bonferroni–Dunn test [33] to detect the significant difference for the ADEA algorithm.

Bonferroni–Dunn's graph is depicted in Fig. 5 to study the significant difference between ADEA and the consulted state-of-art algorithms, for all the benchmark problems.

A horizontal line at $\alpha = 0.05$ and $\alpha = 0.10$ is drawn to show two levels of significance. Critical difference (CD) is computed using equation (13):

$$CD = Q_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{13}$$

where Q_α indicates the critical value for a multiple non-parametric comparison with a control [34]; k and N presents the number of algorithms and the problems taken for comparisons respectively. The ranking and critical difference calculated through Freidman's and Bonnferroni-dunn's procedure is presented in Table 5.

Bonferroni-Dunn's test notifies us of the subsequent significant differences with:

- ADEA as control algorithm:

At α = 0.05

ADEA is better than DE – 1, DE – 2, DE – 3, jDE and the performance of ADEA is at par with jDE and AuDE.

At α = 0.10

ADEA is better than DE – 1, DE – 2, DE – 3, jDE and AuDE.

F	(DE – 1) rand/1/bin (0.9,0.9)		(DE – 2) rand/1/bin (0.5,0.9)		(DE – 3) best/1/bin (0.6,0.3)		ADEA	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
P_1	1.19E+00	8.96E-01	1.35E-71	1.84E-71	**0.00E+00**	0.00E+00	1.93E-77	4.01E-78
P_2	4.10E+03	1.93E+03	**4.95E-12**	6.41E-12	4.94E+01	2.76E+01	**3.45E-12**	5.38E-12
P_3	6.73E-02	1.82E-02	2.25E-03	4.84E-04	2.12E-03	6.42E-04	**2.62E-04**	3.76E-04
P_4	2.44E+03	1.92E+03	1.28E+01	9.16E+00	2.06E+01	3.65E+00	**6.13E-01**	1.76E+00
P_5	6.32E-01	3.79E-01	**3.11E-15**	0.00E+00	7.51E-15	2.08E-15	**5.97E-17**	0.00E+00
P_6	8.31E-01	1.38E-01	5.92E-04	2.01E-03	2.17E-03	3.96E-03	1.87E-04	4.51E-04
P_7	1.27E+02	3.72E+01	1.22E+01	4.11E+00	9.39E+00	3.23E+00	**2.17E-02**	3.34E-01
P_8	5.79E+03	1.40E+03	7.16E+02	7.96E+02	1.99E+02	1.20E+02	**2.56E-05**	1.94E-07
P_9	1.57E+00	2.41E-01	1.82E-01	3.66E-02	1.96E-01	1.96E-02	**8.61E-01**	2.03E-02
P_{10}	7.04E+06	3.17E+07	2.71E+02	1.42E+02	**2.14E+02**	1.27E+02	2.18E+02	4.28E+01
P_{11}	2.12E+00	7.44E-01	**0.00E+00**	**0.00E+00**	**0.00E+00**	0.00E+00	**0.00E+00**	0.00E+00
P_{12}	2.29E-01	2.89E-01	4.15E-03	2.03E-02	**1.57E-32**	4.86E-40	3.83E-29	9.42E-37
P_{13}	1.35E+00	1.42E+00	4.40E-04	2.15E-03	**1.35E-32**	2.60E-40	1.41E-32	1.98E-40

Table 3. Compiled comparative results of 13 benchmark problems (P) on different strategies of DE and ADEA for 30 dimensions

F	jDE		AuDE		ADEA	
	Mean	Std	Mean	Std	Mean	Std
P_1	3.07E-74	7.57E-74	**3.51E-84**	5.92E-84	1.93E-77	4.01E-78
P_2	9.39E-04	9.74E-04	1.14E-09	1.10E-09	**3.45E-12**	5.38E-12
P_3	2.98E-03	8.14E-04	1.43E-03	4.49E-04	**2.62E-04**	3.76E-04
P_4	1.49E+00	3.27E+00	9.57E-01	1.70E+00	**6.13E-01**	1.76E+00
P_5	3.11E-15	0.00E+00	3.11E-15	0.00E+00	**5.97E-17**	0.00E+00
P_6	**0.00E+00**	0.00E+00	8.87E-04	3.03E-03	1.87E-04	4.51E-04
P_7	3.98E-02	1.95E-01	2.20E+01	2.08E+00	**2.17E-02**	3.34E-01
P_8	3.82E-04	7.28E-12	1.10E+02	1.35E+02	**2.56E-05**	1.94E-07
P_9	1.96E-01	1.96E-02	**1.80E-01**	3.99E-02	8.61E-01	2.03E-02
P_{10}	**2.17E+02**	8.19E+01	2.74E+02	5.54E+01	2.18E+02	4.28E+01
P_{11}	**0.00E+00**	0.00E+00	**0.00E+00**	0.00E+00	**0.00E+00**	0.00E+00
P_{12}	**1.57E-32**	4.86E-40	**1.57E-32**	4.86E-40	3.83E-29	9.42E-37
P_{13}	**1.35E-32**	2.60E-40	**1.35E-32**	2.60E-40	1.41E-32	1.98E-40

Table 4. Compiled comparative results of 13 benchmark problems (P) for 30 dimensions

Algorithm	*Mean Rank*
DE – 1	5.15
DE – 2	3.23
DE – 3	2.77
jDE	2.38
AuDE	2.46
ADEA	1.77
CD for α = 0 .05	2.746
CD for α = 0 .10	2.214

Table 5 Ranking and critical difference calculated through Friedman's and Bonnferroni-Dunn's Procedure.

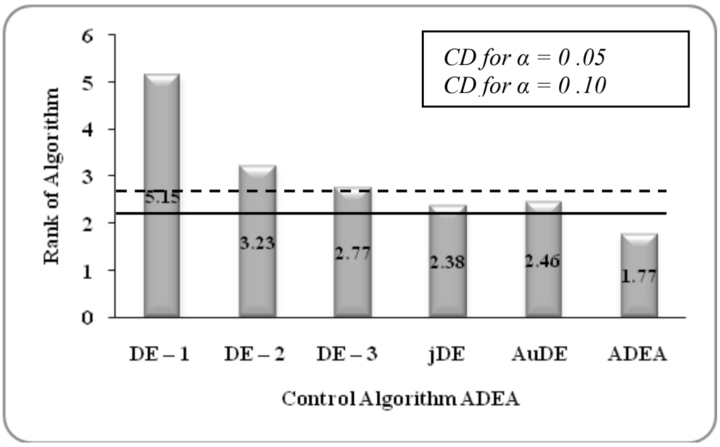


Figure 5. .Bonferroni-Dunn's graphic corresponding to error

B. Noise identification in Multi Noise Plant

Two cases are taken to validate the efficiency of ADEA, the proposed algorithm. The experimental results for DE with different control parameter values in terms of fitness value (best, worst & average), location and accuracy are illustrated in Table 6 – Table 9. The result comparison between DE, GA and ADEA is presented in Table 10. In Table 10 the average result obtained by DE with F = 0.5 & Cr =0.9 are taken for comparison.

Case I: M = 1 (N = 18 & 36)

—*Results in terms of fitness value*

From Table 6, when N = 18, it can be observed that the average fitness values obtained by DE at different control parameters values are 0.04910, 0.03540, and 0.20900 where as the average fitness achieved by ADEA is 0.03427 which is significantly better than DE. Further the best fitness value obtained by ADEA is also comparatively better than DE.

From Table 10, that illustrates the result comparison between DE, GA and ADEA, it can be analyzed that the results achieved by ADEA are significantly better than both the algorithm.

Similarly in case of N = 36, ADEA obtained minimum average fitness value as well the best fitness value in comparison to DE and GA.

—*Location and accuracy*

Using 18 point monitoring system the average location (Loc) and Sound pressure level (SWL in db(A)) obtained by GA and DE are [(4.926, 7.485, 2.123),105.189 db(A)] and [(4.998,7.484,2.00),105.00 db(A)] respectively where as average location (Loc) and Sound pressure level (SWL in db(A)) obtained by ADEA is (4.99950,48390,1.99907), 105.000 db(A)] which is much close to actual location. Similar case can be seen for 36 point monitoring system in Table 10.
Accuracy of the location is also analyzed and reported in Table 7. Δr and ΔSWL represents the distance from actual location and difference from actual SWL respectively. The result demonstrates efficiency of ADEA in comparison to GA and DE.

Case I: M = 2 (N = 18 & 36)

—*Results in terms of fitness value*

The simulated results are illustrated in Table 8 and 9. Table 10 reports the comparative results for GA, DE and ADEA. In this case also, similar to previous one experiment is performed on DE with various combinations of control parameter values. DE with F = 0.5 & Cr = 0.9 control parameter values is carried for result comparisons as best results are obtained at this combination.
From Table 10, in case of both 18 and 36 monitoring points the results obtained by ADEA are significantly better than GA and DE in terms of average location as well as sound pressure level.

—*Location and accuracy*

In this case also, location of Equipment 1 (EQ – 1) and Equipment 2 (EQ – 2) and sound pressure level obtained using ADEA is improved than GA and DE for 18 as well as 36 monitoring points. The results are closer to the actual ones.

N	Parameters		DE			ADEA
			Cr = 0.5, F = 0.5	Cr = 0.9 F = 0.5	Cr = 0.9, F = 0.8	
18	Fitness	Best	0.01390	0.00376	0.11800	0.00302
		Worst	0.05610	0.04120	0.32000	0.04092
		Avg	0.04910	0.03540	0.20900	0.03427
	Std. Dev.		0.01640	0.01080	0.07930	0.07260
	Loc(x,y,z)		4.99500	4.99800	5.02400	4.99950
			7.45700	7.48400	7.46500	7.48390
			1.99000	2.00000	1.94300	1.99907
	SWL db(A)		104.99300	105.00000	104.97600	105.000
36	Fitness	Best	0.01380	0.01110	0.11300	0.01109
		Worst	0.05740	0.04750	0.32900	0.04796
		Avg	0.03110	0.02710	0.19800	0.02699
	Std. Dev.		0.01760	0.01130	0.10700	0.01041
	Loc(x,y,z)		4.99200	5.00000	4.99500	5.00000
			7.49200	7.49800	7.51500	7.50000
			2.01100	2.00100	1.57000	2.00000
	SWL db(A)		104.99500	104.99800	105.02300	105.0000

Table 6 Experimental results for M=1, N=18 & 36 in terms of fitness value (Best, Worst, Average – Avg.), Standard Deviation (Std. Dev.) and average location (x, y,z,), SWL).

MP	Accuracy	GA	DE	ADEA
18	Δr	0.6132	0.339	0.025
	ΔSWL db(A)	0.05	0.025	0.009
36	Δr	0.2425	0.203	0.015
	ΔSWL db(A)	0.219	0.004	0.003

Table 7 Results in terms of accuracy with respect to various monitoring system for M = 1

N	Parameters		DE			ADEA
			Cr = 0.5, F = 0.5	Cr = 0.9 F = 0.5	Cr = 0.9 F = 0.8	
18	Fitness	Best	2.44600	0.14900	9.33700	0.14200
		Worst	7.92000	0.58200	13.52100	0.61733
		Avg	4.53400	0.34500	12.22600	0.33871
	SD		4.10200	0.12200	3.88900	0.10418
	$Loc(x_1,y_1,z_1)$		5.39100	5.00900	5.04400	5.0001
			7.54500	7.58200	7.03600	7.5100
			1.69900	1.88800	1.80000	1.9022
	SWL_1 db(A)		105.27100	104.99700	105.66100	105.0010
	$Loc(x_2,y_2,z_2)$		18.31100	18.00000	18.57200	18.0000
			7.88900	7.58500	6.03200	7.5001
			2.22300	2.91300	1.00000	2.7103
	SWL_2 db(A)		101.74700	102.00000	100.25600	102.0000
36	Fitness	Best	1.26600	0.11900	7.45100	0.11891
		Worst	7.82080	0.41900	12.38600	0.39908
		Avg	3.23220	0.25200	9.35000	0.24954
	SD		3.12010	0.35000	4.72200	0.34092
	$Loc(x_1,y_1,z_1)$		5.18100	5.02700	4.99300	5.0111
			7.51500	7.49700	7.45400	7.5010
			1.79800	1.96200	1.98400	1.9992
	SWL_1 db(A)		105.07200	105.02300	104.95000	105.000
	$Loc(x_2,y_2,z_2)$		18.03100	18.00000	18.25700	18.000
			7.61300	7.55200	7.64000	7.5133
			2.63400	2.97500	1.34500	2.4603
	SWL_2 db(A)		101.55200	101.99800	100.20000	102.0001

Table 8 Experimental results for M=2, N=18 & 36 in terms of fitness value (Best, Worst, Average – Avg.), Standard Deviation (Std. Dev.) and average location (x, y,z,), SWL).

MP	Accuracy	GA	DE	ADEA
18	Δr_1	0.1961	0.6175	0.5732
	Δr_2	1.0712	0.1216	0.0836
	ΔSWL_1 db(A)	0.463	0.024	0.019
	ΔSWL_2 db(A)	1.227	0.032	0.022
36	Δr_1	0.4817	0.0467	0.0361
	Δr_2	0.3849	0.0576	0.0493
	ΔSWL_1 db(A)	0.106	0.023	0.019
	ΔSWL_2 db(A)	0.486	0.002	0.0019

Table 9 Results in terms of accuracy with respect to various monitoring system for M = 2

Plant	Item	Algorithm	N = 18	N = 36
M = 1	Fitness Value	GA	0.4620	0.0989
		DE	0.0354	0.0271
		ADEA	0.03427	0.02699
	Loc (x,y,z) & SWL db(A)	GA	(4.926, 7.485,2.123, 105.189)	(5.009, 7.426,1.985, 105.05)
		DE	(4.998, 7.484, 2.00, 105.00)	(5.00, 7.498,2.001, 104.998)
		ADEA	(4.99950, 48390, 1.99907, 105.000)	(5.00000, 7.50000, 2.00000, 105.0000)
M = 2	Fitness Value	GA	0.89800	0.59800
		DE	0.34500	0.25200
		ADEA	0.33871	0.24954
	Loc (x_1,y_1,z_1) & SWL_1 db(A)	GA	(4.814, 7.123, 1.451, 104.541)	(4.951, 7.462, 2.012, 104.818)
		DE	(5.009, 7.582,1.888, 104.997)	(5.027, 7.497, 1.962, 105.023)
		ADEA	(5.0001, 7.5100, 1.9022, 105.0010)	(5.0111, 7.5010,1.9992, 105.000)
	Loc (x_2,y_2,z_2) & SWL_2 db(A)	GA	(17.250, 7.601, 2.012, 101.5433)	(17.978, 7.709,2.994, 102.273)
		DE	(18.000,7.585, 2.913, 101.9980)	(18.00, 7.552, 2.975, 101.998)
		ADEA	(18.0000, 7.5001, 2.7103, 102.000)	(18.000, 7.5133,2.4603, 102.0001)

Table 10 Comparative experimental results for M=1, N=18 & 36 in terms of fitness value (Best, Worst, Average – Avg.), Standard Deviation (Std. Dev.) and average location (x, y,z,), SWL).

VII. Conclusions and Future Scope

An improvement in mutation process of basic DE is introduced in this study with an intension of accelerating the convergence rate. The concept of reflection is used to produce new decorative positions. This phenomenon is taken from Interior search algorithm. The mirror is placed near the most beautiful (global best) individual to accentuate their attractiveness. This new variant is named as aesthetic differential evolution algorithm (ADEA). ADEA is implemented on thirteen benchmark problems discussed in CEC 2006. Non-parametric test are performed to evaluate the significance of the proposal. Further to investigate the efficacy of the proposal a case study of identification of noise in multi plant is solved. In first case single equipment with 18 and 36 monitoring point is considered where as in second case two equipments with 18 and 36 monitoring point is considered. The proposal is consistently able to reach the optimal solutions in terms of best fitness value, average value as well as in identifying locations with better sound level pressure. The computed statistical results indicate significant difference in terms of performance of the proposal with respect to state-of-art algorithms.

In future ADEA will be implemented on constrained optimization real world problems.

Acknowledgment

Authors are thankful to the reviewers for their valuable suggestions in order to improve the quality of the paper. Also, we are thankful to the Organizers of NaBIC 2015 for giving us the opportunity to present the extended work in this journal of repute.

References

[1] R. Storn, K. Price. “Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces”. *Tech. Rep. TR-95-012*, ICSI, USA, 1995.

[2] R. Storn, K. Price. “Differential evolution - A simple and efficient heuristic for global optimization over continuous spaces”, *Journal of Global Optimization*, 11(4), pp. 341-359, 1997.

[3] M. M. Ali, A. Torn. “Population set based global optimization algorithms: Some modifications and numerical studies”, *Comput. Oper. Res.*, 31(10), pp. 1703-1725, 2004.

[4] J. Brest, S. Greiner, B. Bovskovic, M. Mernik, V. Zumer. “Self adapting control parameters in differential evolution: A comparative study on numerical benchmark problems”, *IEEE Transactions on Evolutionary Computation*, 10(6), pp. 646-657, 2006.

[5] J. Lampinen, I. Zelinka. "On stagnation of the differential evolution algorithm". In *Proceedings of the. 6th International Mendel Conf. Soft Computing*, pp. 76–83, 2002.

[6] R. Gämperle, S. Müller, P. Koumoutsakos. "A parameter study for differential evolution". In *Proceedings of the Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation*, pp. 293–298, 2002.

[7] A.Singh Poonia , T. K. Sharma , S. Sharma , J. Rajpurohit. "Aesthetic Differential Evolution Algorithm for Solving Computationally Expensive Optimization Problems". In *Proceedings of Advances in Nature and Biologically Inspired Computing*, Volume 419, pp 87-96, 2016.

[8] P. Kumar, M. Pant. "Noisy source recognition in multi noise plants by differential evolution," In *Proceedings of 2013 IEEE Swarm Intelligence Symposium (SIS)*, pp. 271 – 275, 2013.

[9] A. Ghosh, S. Das , A. Chowdhury, R. Giri. "An improved differential evolution algorithm with fitness-based adaptation of the control parameters", *Information Sciences*, 181(18), pp. 3749-3765, 2011.

[10] D. Zou, H. Liu, L. Gao, S. Li. "An improved differential evolution algorithm for the task assignment problem", *Engineering Applications of Artificial Intelligence*, 24(4), pp. 616-624, 2011.

[11] L. Wang, X Fu, Y. Mao, M. I. Menhas, M. Fei. "A novel modified binary differential evolution algorithm and its applications", *Neurocomputing*, 98(3),pp. 55-75, 2012.

[12] A. W. Mohamed, H. Z. Sabry. "Constrained optimization based on modified differential evolution algorithm", *Information Sciences*, 194, pp. 171-208, 2012.

[13] D. Zou, J. Wu, L. Gao, S. Li. "A modified differential evolution algorithm for unconstrained optimization problems, *Neurocomputing*, 120, pp. 469-481, 2013.

[14] W. Deng, X. Yang, L. Zou, M. Wang, Y. Liu, Y. Li. "An improved self-adaptive differential evolution algorithm and its application", *Chemometrics and Intelligent Laboratory Systems*, 128, pp. 66-76, 2013.

[15] M. Ali, C. W. Ahn, P. Siarry. "Differential evolution algorithm for the selection of optimal scaling factors in image watermarking", *Engineering Applications of Artificial Intelligence*, 31, pp. 15-26, 2014.

[16] Coelho L.dos S., Ayala HVH., Mariani VC. "A self-adaptive chaotic differential evolution algorithm using gamma distribution for unconstrained global optimization", *Applied Mathematics and Computation*, 234, pp. 452-459, 2014.

[17] A.W. Mohamed. "An improved differential evolution algorithm with triangular mutation for global numerical optimization", *Computers & Industrial Engineering*, 85, pp. 359-375, 2015.

[18] Xiang Wan-li, Zhu N., Ma Shou-feng, Meng Xue-lei, An Mei-qing. "A dynamic shuffled differential evolution algorithm for data clustering", *Neurocomputing*, 158, pp. 144-154, 2015.

[19] R. Mallipeddi, M. Lee. "An evolving surrogate model-based differential evolution algorithm", *Applied Soft Computing*, 34, pp. 770-787, 2015.

[20] A. Zamuda, J. Brest. "Self-adaptive control parameters′ randomization frequency and propagations in differential evolution", Swarm and Evolutionary Computation, 25, pp. 72-99, 2015.

[21] Y. Cai, J. Wang. "Differential evolution with hybrid linkage crossover", Information Sciences, 320, pp. 244-287, 2015.

[22] X. Wang, L. Tang. "An adaptive multi-population differential evolution algorithm for continuous multi-objective optimization", Information Sciences, 348, pp. 124-141, 2016.

[23] M. Basu. "Quasi-oppositional differential evolution for optimal reactive power dispatch", International Journal of Electrical Power & Energy Systems, 78, pp. 29-40, 2016.

[24] W. Yi, Y. Zhou, L. Gao, X. Li, J. Mou. "An improved adaptive differential evolution algorithm for continuous optimization", Expert Systems with Applications, 44, pp. 1-12, 2016.

[25] Mashwani WK. "Enhanced versions of differential evolution: state-of-the-art survey", *Int. J. Computing Science and Mathematics*, 5(2), pp.107–126, 2014.

[26] Gandomi AH. "Interior Search Algorithm (ISA): A Novel Approach for Global Optimization", *ISA Transactions*, 53(4), pp. 1168–1183, 2014.

[27] Gandomi AH., Yang XS. "Evolutionary Boundary Constraint Handling Scheme", *Neural Computing & Applications*, 21(6), pp. 1449-1462, 2012.

[28] E. Mezura-Montes, J. Velazquez-Reyes, C. A. Coello Coello. "A comparative study of differential evolution variants for global optimization," In *Proceedings Genet. Evol. Comput. Conf.*, pp. 485- 492, 2006.

[29] Ji Qiang, C. Mitchell. "An Adaptive Unified Differential Evolution Algorithm for Global Optimization" *LBNL Report Number: LBNL-6853E*, 2014.

[30] B. David et Hansen, Colin H. *Engineering Noise Control, Theory and practice*, Unwin Hyman, London, 1988.

[31] J. Demšar. "Statistical comparisons of classifiers over multiple data sets", *J Mach Learn Res*, 7, pp. 1–30, 2006.

[32] S. García, F. Herrera. "An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons", *J Mach Learn Res*, 9, pp. 2677–2694, 2008.

[33] OJ. Dunn. "Multiple comparisons among means", *J Am Stat Assoc*, 56(293), pp. 52–64, 1961.

[34] JH. Zar. *Biostatistical analysis*, NJ:Prentice-Hall, Englewood Cliffs, 1999.

Tarun Kumar Sharma MCA from SGRRITS Dehradun, MBA in Marketing from VMOU Kota and PhD in Computational Intelligence from Indian Institute of Technology Roorkee, India is presently affiliated with Amity University Rajasthan, Jaipur, India. His area of interest is developing hybridized nature inspired algorithms and their application in versatile engineering domains. He has published several papers in international journals and conferences of repute.

Millie Pant received M.Sc in Mathematics from CCS University Meerut and PhD from Indian Institute of Technology Roorkee, India presently affiliated as Associate Professor in the Department of Applied Science and Engineering. She has supervised 8 PhD's. Her expertise is in designing hybrid variants of Evolutionary algorithms. Her area of interest comprises of Numerical Optimization, Evolutionary Algorithms.