Journal of Marine Science and Technology
DOI: 10.6119/JMST-012-0917-1

This article has been peer reviewed and accepted for publication in JMST but has not yet been copyediting, typesetting, pagination and proofreading process. Please note that the publication version of this article may be different from this version. This article now can be cited as doi: 10.6119/JMST-012-0917-1.

A Creative Differential Evolution Algorithm for Global Optimization Problems

Hsin-Chuan Kuo[1], Ching-Hai Lin[1], and Jeun-Len Wu[1]

Key words: Global optimizations, Differential evolution algorithms, Garbage can decision-making model, Creative differential evolution algorithm.

Abstract

Under the framework of the differential evolution algorithm, we develop a creative algorithm by extrapolating the thinking of the garbage can model. The developed algorithm is hence named in the study as a creative differential evolution algorithm (CDE).

To verify the performance of the CDE, we selected seven well-known benchmark functions; three of them are uni-modal and four multi-modal. In conducting the numerical experiment, we adopted two different numbers of dimensions for each test functions, which is 50 and 100. The results show that CDE can find the global optimum robustly, demonstrating that CDE significantly improves the DE's performance.

I. INTRODUCTION

The mathematical model for an optimization problem is

$$Minimize \quad f(X) = f(x_1, x_2, ..., x_n) \qquad (1)$$

Where $x_1, x_2, ..., x_n$ are design variables, and n is the dimensionality. Using the mathematical programming method of differential derivatives, the local optimum to the problem can efficiently be found. If the differential derivative of the problem cannot be easily obtained, a direct search approach may be used without requiring the calculation of the derivatives, e.g., Nelder and Mead's Simplex method [15]. It is hard to find the global optimum of multi-modal problems by mathematical programming methods because it is difficult to determine whether the obtained solution is the global optimum.

Since the 70s, the use of biological concepts such as genetics, survival of the fittest, and group behavior have been popular in the study of optimization methods. Holland [10] first proposed the Genetic Algorithm (GA) in 1975 to imitate Darwin's biological evolution model of natural selection and survival of the fittest. Dorigo [7] proposed the Ant Colony Optimization (ACO) in 1991, which imitated an ant colony foraging. The ant colony could distinguish the shortest path to the food by the intensity of the pheromone left on the path. This idea can be applied to solve many combinatorial optimization problems. Kennedy and Eberhart [11] proposed the Particle Swarm Optimization (PSO) algorithm in 1995 by mimicking the migration of birds. Storn and Price [19] proposed the Differential Evolution Algorithms (DEs) in 1997.

Storn and Price [19] stated, "DE borrows the idea from Nelder & Mead's Simplex method of employing information from within the vector population to alter the search space. DE's self-organizing scheme takes the difference vector of the randomly chosen population vectors to perturb an existing vector. The perturbation is done for every population vector." Later, Feokitistov and Janaqi [8] compared DEs and GAs; although both use a population search to find the optimum, the major difference between the two is in the mutations. In GAs, a small perturbation is applied to individual genes, whereas in DEs, arithmetic combinations are employed between individuals. The main components of DE are similar to those of GA, such as population system, selection scheme, mutation operator, and crossover operator. Storn and Price [19], in 1997, provided several mutation strategies for user's need. Currently, DE is considered one of the most reliable, accurate, robust, and quickly convergent optimization algorithms available and is widely applied in numerous fields: Cheng *et al.* [3] in 2001 applied DE to cope with linear system models; in 2002, Abbass [1] used DE and the artificial neural networks approach for breast cancer diagnosis of medical science; Cheong *et al.* [4], in 2007, used DE to design a hierarchical fuzzy logic controller for controlling a cart-pole with four state variables. However, DE's parameters are problem-dependent and it is costly to find the best parameters. Furthermore, DE's performance is quite sensitive to the values of these parameters [31].

Up to now, many strategies on improving the DE's performance have been developed since introduced, and herein some of them are briefed as follows:

Ali *et al.* [2] in 2004 proposed a rule for calculating the control parameter F; Qin *et al.* [18], in 2005, proposed a strategy of self-adapting population size NP, mutation parameter F, and crossover rate CR for reducing the sensitivity of control parameters. Teo [24] in 2005 and Nobakhti *et al.* [16] in 2008 proposed a self-adapting mechanism and a suitable learning strategy to control the parameters F and CR. Sun *et al.* in 2005 [20] proposed a hybrid approach combining DE and EDA (Estimation of Distribution Algorithms) for global continuous optimization problems. DE/EDA combines global information (i.e., distance and direction information) extracted

---

Paper submitted 08/16/11; revised 04/10/12; accepted 09/17/12. Author for correspondence: Hsin-Chuan Kuo (e-mail: khc@ntou.edu.tw).
[1]Department of System Engineering and Naval Architecture, National Taiwan Ocean University, Keelung, Taiwan, R.O.C.

by EDA with differential information obtained by DE to create promising solutions. Xua *et al*. [28] in 2007 proposed a hybrid of DE and Particle Swarm Optimization (DEPSO) in training recurrent neural networks (RNNs). Kaelo *et al*. [12] in 2006 suggested modifications in mutation and localization in acceptance rule to the differential evolution algorithm for global optimization. Chen *et al*. in 2008 [5] proposed a refreshing distribution operation into solution search process, i.e., a number of excellent individuals are preserved, and the rest individuals will be initialized randomly to maintain the population diversity. Yang *et al*. [29] in 2008 introduced a sharing function method into DE to solve a power system planning problem. Wang *et al*. [26] in 2007 introduced a dynamic clustering technique into DE to improve the performance of DE when applied to global optimization problems. The modified DE algorithm, during the search process, gradually changes from exploring promising areas at the early stages to exploiting solution with high precision at the later stages.

The DE performance-enhancing strategies proposed by the above literatures are found with three main directions. The first focuses on adjusting the control parameters NP, F, and CR; the next combines DE with other algorithms to form a hybrid model of DE; and the last introduces a search strategy to improve the solution precision. Noticeably, the shape of a problem's contour lines around its exact solution is found closely associated with the improvement of DE.

The improvement is more obvious for a sphere than a rotated ellipse. For example, the Rosenbrock problem, a well-known benchmark test function, possesses long and narrow band contour, which makes securing a solution close enough to the function's exact solution much more challenging using varied strategies. Especially for problems with high dimensions, getting a reliable solution using DE becomes nearly impossible.

In regard to multi-modal problems which possess many local extremities, in the evolution process individuals locating at the regions of the local extremities are easy to be trapped within and then have troubles to escape from the regions in the upcoming evolution. Many versions of DE have been studied to enhance the individual's exploration ability for these multi-modal problems. However, the improvement is more obvious for a problem with a noticeable difference of values between local and global extremities than the other with extremities having almost the same values, such as the Generalized Schwefel's Problem 2.26 [30]. Besides, as the number of the dimensions of a problem, uni-modal or multi-modal, is increasing, the difficulty in getting the closest solution to the exact one is becoming harder and harder, even out of the question to achieve it.

The poor improvement of these variants of DEs for the scenarios described above is probably attributed to their only considering the interactions among the individuals but ignoring individuals' evolution experiences. However, this paper shows that the population evolution experiences, if collected and systematically analyzed, will provide helpful information on finding better solutions with less effort.

In the 1980s, Maynard Smith proposed evolutionary game theory based on the classic game theory [21]. Liu *et al.* [13] applied it to a particle swarm optimization algorithm. Wiegand *et al.* [27] used the model to analyze the dynamics of co-evolutionary algorithms. In 1960, Simon [22] proposed Intelligence, Design and Choice, a three-stage repetitive decision process. In 1972, Cohen *et al.* [6] proposed the Garbage Can Model, a decision model for organized anarchies. The characteristics of the problematic preference and unclear technology described in the Garbage Can Model are found and existed in an organized anarchy.

In the work, we named the to-be-developed variant of the differential evolution algorithm, inspired by the Garbage Can Model, as the creative differential evolution algorithm (CDE). The search procedure of the CDE will be presented in the next paragraph.

To verify the performance of the CDE, we applied it in the study to seven benchmark functions, three of which are uni-modal, and four multi-modal. The number of dimensions of each test function is made as high as a value of 100, and the results obtained are to be compared with those by other studies and discussed.

II. Differential Evolution Algorithm (DE)

In 1997, Storn and Price [19] proposed a population-based evolutionary algorithm that operated competitions among individuals of different generations. This algorithm adopts the search frame and the gene mutation and crossover operators of the classical GA. The important step for the mutation operation of DE is a new scheme different from GA in generating trial vectors. DE generates new trial vectors by adding the weighted difference vector between two population individuals to a third individual.

The main steps of the DE algorithm are given below:

Initialization
Evaluation
Repeat
Mutation
Crossover
Evaluation
Selection
Until *(termination criteria are met)*

In the evolution process, the increment of mutations, i.e., the search step, makes the populations explore and exploit, compete with each other, and then finally achieves the optimal solution. This process shows that in the beginning of the evolution, the mutation operator guides the exploration of the population in the search space. Subsequently, the mutation operator makes the population coalesce to the best individual and then exploits the population. There are three basic steps to create a new generation: mutation, crossover, and selection:

(1) Mutation

Mutation is carried out by the trial vector. Roughly speaking, it is the target vector plus the product of the mutation factor F and the difference vector. The difference vector is obtained by taking the difference of two or more random vectors. The base vector is X_{r1}^G (as shown in Fig. 1) and the difference vector is the difference between X_{r2}^G and X_{r3}^G (subtraction). The three vectors are all randomly selected individuals.

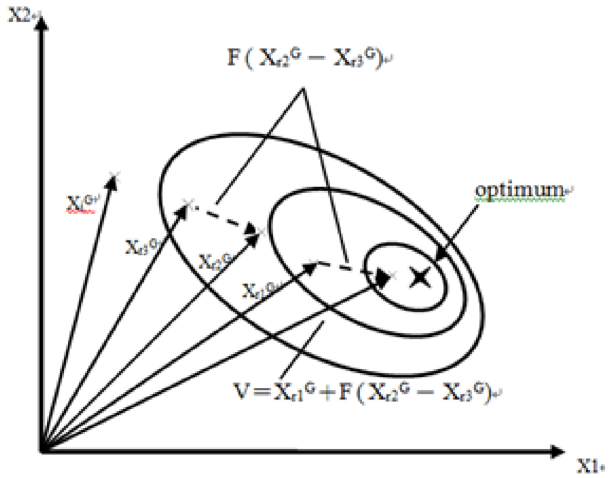

Fig. 1. The generation of a trial vector in a mutation[19].

The mutation operation for generating the trial vector V^{G+1} is as follows:

$$V^{G+1} = X_{r1}^G + F(X_{r2}^G - X_{r3}^G) , \tag{2}$$

where F is the mutation parameter between (0, 2), and $r1, r2$ and $r3$ are different individuals randomly selected from the *G*-th generation. X_{best} is the best individual of the *G*-th generation.

DE mutation strategies [19]:

a. DE/best/1： $X_{best} + F(X_{r1} - X_{r2})$

(3)

b. DE/rand/1： $X_{r1} + F(X_{r2} - X_{r3})$ **(4)**

c. DE/rand-to-best/1：

$$X_{r1} + F(X_{best} - X_{r1}) + F(X_{r2} - X_{r3}) \tag{5}$$

d. DE/best/2： $X_{best} + F(X_{r1} + X_{r2} - X_{r3} - X_{r4})$ (6)

e. DE/rand/2： $X_{r5} + F(X_{r1} + X_{r2} - X_{r3} - X_{r4})$ (7)

In the notation DE/*a*/*m*/z used above, *a* represents the individual being perturbed and *m* is the number of difference vectors to perturb *a*, z = bin, and is henceforth neglected in this paper.

From the above list, the user should choose a single mutation strategy that fits the problem to create mutation vectors.

The individual i in the *G*-th generation is:

$$X_i^G = (x_{1i}^G, x_{2i}^G, \ldots, x_{ni}^G) \tag{8}$$

The trial vector is:

$$V^{G+1} = (v_1^{G+1}, v_2^{G+1}, \ldots, v_n^{G+1}) \tag{9}$$

(2) Crossover

Crossovers are also carried out randomly.

The child *y* created by crossover between mother generation X_i^G and V^{G+1} is：

$$y = (y_1, y_2, \ldots, y_n) \tag{10}$$

$$y_j = \begin{cases} V_j^{G+1}, & \text{if } r_j \le CR \text{ or } j = l \\ X_{ji}^G, & \text{if } r_j > CR \text{ and } j \ne l \end{cases} \tag{11}$$

where $j = 1,2,3,\ldots,n$; l is a random integer such that $l \in \{1,2,\ldots,n\}$; and r is a random number such that $r_j \in U(0,1)$. The crossover rate (CR) must be satisfied the $0 \le CR \le 1$.

(3)Evaluation/ Selection

The individual X_i^G and the child *y* created by mutation and crossover compete with each other. The good individual X_i^{G+1} is considered as the children in the next generation.

$$(X_i^{G+1}) = \arg\min\{F(X_i^G), F(y)\} \tag{12}$$

Then the best individual in the next generation is

$$(X_{best}^{G+1}) = \arg\min\{F(X_i^G)\} , i=1,..,NP \tag{13}$$

where $F(X)$ is the objective function (the fitness measure), NP is population size.

III. Garbage Can Decision-Making Model

Decision making is an important task in an organization. There are three theories about the decision-making process:

(1) Classical Model:

This model, proposed by Taylor [25] in 1947, is an optimistic strategy, assuming that decision-makers are fully rational and use the most promising method to achieve their goals. Their results are perfect.

(2) Administrative Model:

Proposed by Simon [23] in 1974. This model is a satisficing strategy, assuming that it is impossible for decision-makers to make perfect decisions because they are limited by the existing knowledge, resources and information. They can only try to achieve satisfactory results.

(3) Incremental Model:

Proposed by Lindblom [14] in 1959. This model approximates a constant comparison strategy under a certain threshold. It assumes that because of the limitations of existing knowledge, resources and information, decision-makers can make neither perfect nor satisfactory decisions. They identify the feasible decision most suitable for the current situation only through constant comparison. In this model, the decision-makers can be considered aware of their goals, but uncertain of the actions required to achieve them. They assess the results after reaching an intermediate stage and adjust the direction in which they advance. The model is very similar to the real decision making process in a human social organization.

Based on Simon's bounded rationality theory [22], Cohen et al. argued that decision-makers, limited both by inadequate external information and by subjective human feelings that impair rational judgments, can never make the perfect decision. In 1972 [6], Cohen et al. proposed that the organization of an actual decision-making process is best modeled in an entirely different way: the Organized Anarchies Model is resembles the evolution model. In this model, the decision-making process of an organization is like an evolving process. This model has three characteristics:

(1) Problematic Objective Preferences

Decision-makers have inconsistent preferences concerning the problems and goals, and these preferences can only be discovered through actions. Therefore, these preferences cannot be the basis for actions.

(2) Unclear Technology

Group members only know that something needs to be improved during decision-making, but they do not know what individuals should do to improve. Therefore, they have to use trial and error methods based on their personal knowledge.

(3) Fluid Participation

If the issues are controversial and a lengthy investigation is required before any decision is made, it is quite probable that the final decision-making group will not be comprised of the same people as the initial group. Decision-makers can also come from various perspectives, topics of interest and all walks of life.

In such ambiguous situations, each decision-making process is regarded as a receptacle or garbage can, in which decision-makers, issues, and solutions are represented by garbage. This decision-making model is called the Garbage Can Model (GCM).

IV. Creative Differential Evolution Algorithm (GDE)

In employing the differential evolution algorithm, users have to make a decision of selecting one among the five mutation strategies of the algorithm. The fact that which one is suitable at the current generation for the problem of interest is actually uncertain is similar to the unclear technology as depicted in the garbage can model. Further, in the evolution process each individual in the DE population standing a possible solution to the problem resembles to each member in an organization providing a feasible solution to an encountering problem. Therefore, inspired by the decision policy process of the garbage can model, we attempted to apply the process with the DE algorithm for the purpose of improving the DE's performance.

In the application of the original garbage can model, we redefined the significance of the three different kinds of the model's features, problematic objective preferences, unclear technology and fluid participation, in order to formulate in the study a new algorithm, which is named as Creative Differential Evolution Algorithm (CDE). The definitions of the three features are described as the following.

(1) Open innovation:

One must think outside the box and embrace creativity.

(2) Integrate innovation:

One must overcome disagreements with an integrated objective model.

(3) Process innovation:

One must increase the exchange of ideas and grasp any opportunity to solve the problem.

After several evolution generations, the best individual is the most satisfactory decision that can be made by the participants of the organization.

In this paper a diverse selection model for mutation is proposed that several mutation strategies, being referred to as the "mutation-strategy can", are included in the mutation operation. The mutation strategy for each generation is then randomly selected from this can. This application of the mutation-strategy-can to the differential evolution algorithm will be referred to as Creative Differential Evolution Algorithm with Dynamic Mutation (CDEDM).

When the population matures, the grouping search strategy is used to split the mother population into several child populations (several subpopulations) and integrate the search results of each child population. In other words, when the population matures, the strategy identifies the promising space of each variable of each child population and then searches separately in each promising space. For example, if the population contains 800 individuals, 400 of which are better ones, the promising space is defined by the intervals of each variable of these 400 individuals. Therefore, after every few generations (called the evaluated generation number), the parent population is divided into three child populations and each child population is confined by a search space, as shown in Fig. 2.

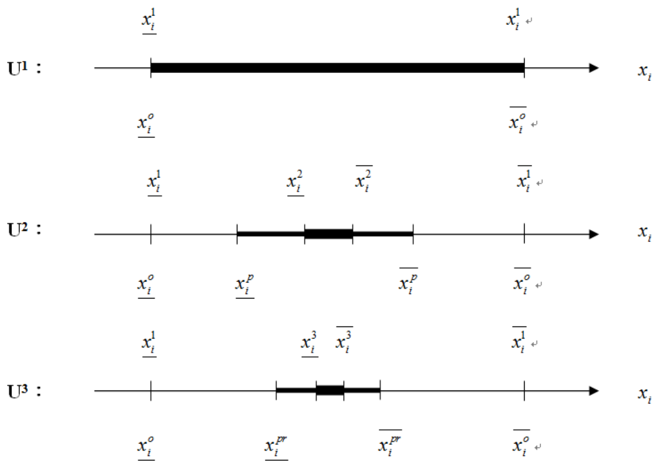

Fig. 2. CDE's grouping search strategy.

(1) The first child population:

The search space U^1 is the initial search space, $U_i^0 = [\underline{x}_i^0, \overline{x}_i^0]$, $i = 1,2,\ldots,n$. $\underline{x}_i^0, \overline{x}_i^0$ are the lower and upper bounds, respectively.

(2) The second child population:

The search space U^2 is the promising space, $U_i^p = [\underline{x}_i^p, \overline{x}_i^p]$, and then is broadened.

(3) The third child population:

The search space U^3 is that the promising space is reduced by the roulette wheel selection model to $U_i^{pr} = [\underline{x}_i^{pr}, \overline{x}_i^{pr}]$ and subsequently broadened. For example, each variable is partitioned into five segments. According to the probability model of a roulette wheel, the individuals from each segment with better average function values are assigned higher probabilities of being selected. Finally, one of the five segments is selected segment so that convergence takes place quickly.

By applying the grouping search strategy in CDEDM, in this paper, another new version of DE, which is called the Creative Differential Evolution algorithm (CDE), is proposed. Its flowchart is shown in Fig. 3.

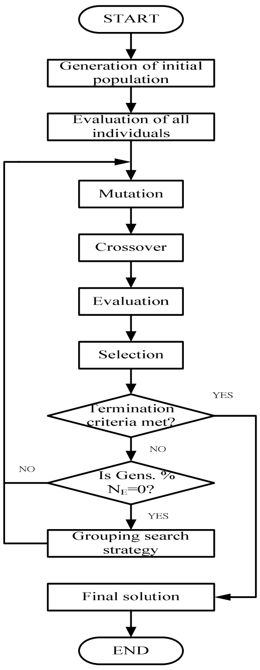

Fig. 3. Flow chart of CDE.

In order to simulate the decision-making behavior of human society, four parameters are set: the evaluated generation number, the number of better individuals, the broaden ratio, and the number of partitioned segments. In this paper, the number of partitioned segments is set to 5. The other parameters are described below:

(1) Evaluated generation numbers, N_E: After the passing of every N_E generations, the grouping search strategy is applied to the evolution process of population to simulate regular group meeting.

(2) Number of better individuals, N_B: In the population, the individuals with better function values are selected. Their lower and upper bounds are used to confine the promising space, simulating the choices made in a meeting.

(3) Broaden ratio, B_r: Once the promising space is determined, it is broadened, simulating flexible decisions in order to improve the population's optimization performance.

V. Experimental results and discussion

1. Benchmark Functions

In order to test the search performance of CDEDM and CDE, the 3 uni-modal and 4 multi-modal benchmark functions from [31, 26] are examined.

Uni-modal functions：

(1) f_1：*Sphere* function

$$F(X) = \sum_{i=1}^{n} x_i^2 \tag{14}$$

where the global optimum solution is to be $X^* = 0$ and the global optimum objective function is $F(X^*) = 0$ for $-100 \le x_i \le 100$. The sphere function is continuous, differentiable, separable, scalable, uni-modal, and symmetric.

(2) f_2：*Rotated hyper-ellipsoid* function

$$F(X) = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j\right)^2 \tag{15}$$

where $X^* = 0$ and $F(X^*) = 0$ for $-100 \le x_i \le 100$. It is continuous, differentiable, non-separable, scalable, uni-modal, and asymmetric. The effect of symmetry and rotation can be investigated through it.

(3) f_3：*Rosenbrock* function

$$F(X) = \sum_{i=1}^{n-1} [100(x_i - x_{i+1}^2)^2 + (x_i - 1)^2] \tag{16}$$

where $X^* = \{1,1,\ldots,1\}$ and $F(X^*) = 0$ for $-30 \le x_i \le 30$. The Rosenbrock function is continuous, differentiable, non-separable, scalable, uni-modal, and asymmetric.

Multi-modal functions：

(1) f_4：*Ackley's* function

$$F(X) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e \tag{17}$$

where $X^* = 0$ and $F(X^*) = 0$ for $-32 \le x_i \le 32$. The Ackley function is continuous, differentiable, non-separable, scalable, multi-modal, and symmetric. The *Ackley* function has a narrow region of global optimum, with many unremarkable local optima nearby. It is suitable for investigating the effect of modality and separability.

(2) f_5：*Griewank* function

$$F(X) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \tag{18}$$

where $X^* = 0$ and $F(X^*) = 0$ for $-600 \le x_i \le 600$. The Griewank function is continuous, differentiable, non-separable, scalable, multi-modal, and asymmetric. It is difficult to optimize. It is a good problem for studying the effect of modality, separability and symmetry.

(3) f_6 : *Rastrigin* function

$$F(X) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10) \quad (19)$$

where $X^* = 0$ and $F(X^*) = 0$ for $-5.12 \le x_i \le 5.12$. The Rastrigin function is continuous, differentiable, separable, scalable, multi-modal, and symmetric. The effect of modality can be studied through it.

(4) f_7 : *Generalized Schwefel's* problem 2.26

$$F(X) = \sum_{i=1}^{n} x_i \sin(\sqrt{|x_i|}) \quad (20)$$

where $X^* = \{-420.9687, ..., -420.9687\}$ and $F(X^*) = -418.983*n$

for $-500 \le x_i \le 500$. The Schwefel's problem 2.26 is continuous, non-differentiable, separable, scalable, multimodal, and symmetric. The global optimum of the *Generalized Schwefel's* problem 2.26 falls at the bottom left corner of the search space, whereas a comparable local optimum falls at the top right corner. The effect of differentiability and modality can be examined by using it.

2. Parameter settings for the DE, CDEDM, and CDE

Parameters of DE, CDEDM and CDE for the 50- and 100-dimensional benchmark functions are follows:

The mutation strategies are:

In each generation, CDEDM and CDE randomly choose strategy from the strategy can including (1)DE/best/1，(2) DE/rand-to-best/1 ，(3)DE/rand/2; DE uses the mutation strategy DE/best/1.

The parameters are:

A. Dimension *n* : 50 , 100
B. Population size NP : 400 for *n=50*; 800 for *n*=100 [17]
C. Maximal number of generations MaxGen : 4000 for *n=50*; 6000 for *n*=100
D. Mutation parameter F =0.5~ 0.6
E. Crossover rate CR = 0.3 [9]
F. Convergence condition ε= 1.E-05 $\begin{cases} f_1 \sim f_6 \\ f_7 \quad -418.983*n \end{cases}$

The additional parameters for CDE are:

A. Evaluated number of generations : 20 for *n=50*; 200 for *n*=100
B. Number of better individuals : 0.5NP
C. Broaden ratio : 10

The three algorithms are tested on the seven benchmark functions. Each function is independently tested 30 times. f_{av} is the average objective function value, f_b is the best value, f_w is the worst value, and σ_f is the standard deviation.

3. Experimental results and analysis

The DE, CDEDM, and CDE algorithms are tested on the seven benchmark functions of 50 and 100 dimensions. The results are listed in Tables 1 and 2.

It is clear from the Table 1 that for the 50-dimensional functions f_1~f_6, CDE can robustly find the global optimum, while CDEDM has a chance of finding only the global optimum of function f_3 and is unstable overall. For the 50-dimensional function f_7, CDE achieves f_b = -20949.14, which is close to the global optimum. The f_{av} of CDE is in general better than that of CDEDM. These results show that compared with the other two algorithms, CDE's performance is more robust and efficient in finding the global optimum solution of the 50-dimensional benchmark functions. With regard to the search performance of CDEDM and DE, CDEDM is more robust than DE for the uni-modal functions f_1 and f_2, as well as for the multi-modal functions f_4 and f_6, and achieves acceptable results. For the multi-modal functions f_7, CDEDM's f_{av} is slightly better than that of DE by about -134.66. For the function f_5, CDEDM is no clear improvement. CDEDM performs worse than DE for the function f_3.

As shown in Table 2, for the 100-dimensional functions f_1-f_6, CDE's performance is also very reliable and efficient compared to DE and CDEDM. CDEDM's performance on function f_3 does not better than DE's. This is probably because diverse mutation strategies make the population converge slowly. For the 100-dimensional function f_7, CDE achieves f_b = -40951.78, better than both DE and CDEDM. The standard deviation of CDE, which is also the smallest, shows that CDE is superior to the other two algorithms in diversity and intensity.

4. Analysis of the population shift of the DE, CDEDM and CDE in 100-dimensional benchmark functions

In the process of searching and shifting a population, two distance indexes are used to analyze the dynamics of the population shift. One is the distance between all individuals of the population and the optimum, the other is the distance between all individuals and the analytic solution. These two average distances are denoted by d_{pg} (%) and d_{pa} (%), respectively.

The average distance $d_{pg}(\%)$ is defined as follows:

$$d_{pg}(\%) = \frac{\overline{d_{pg}}}{L} \times 100\% \quad (20)$$

where $\overline{d_{pg}} = \frac{\sum_{i=1}^{N_p} \left\| X_i - X^{gbest} \right\|}{N_p}$ is the average distance between the population and the best individual solution X^{gbest} and $L = \sqrt{\sum_{i=1}^{n} \left(\overline{x_i} - \underline{x_i} \right)^2}$, where $\overline{x_i}$ and $\underline{x_i}$ are the lower and the upper bound of the i -th dimension respectively.

Table 1. Results of DE, CDEDM and CDE for 50-dimensional benchmark problems.

Method / Function	CDE			CDEDM			DE		
	f_{av}	f_b / f_w	σ_f	f_{av}	f_b / f_w	σ_f	f_{av}	f_b / f_w	σ_f
f_1 : Sphere	9.03E-06	7.19E-06	5.95E-07	9.10E-06	6.81E-06	5.99E-07	9.41E-06	8.42E-06	6.22E-07
		9.96E-06			9.98E-06			1.00E-05	
f_2 : Ellipsoid	9.34E-06	8.27E-06	4.80E-07	9.35E-06	8.66E-06	4.50E-07	1.08E-05	5.96E-06	5.63E-06
		9.95E-06			9.99E-06			3.12E-05	
f_3 : Rosenbrock	9.00E-06	3.99E-06	1.31E-06	1.99E-01	4.99E-06	8.68E-01	9.35E-06	8.01E-06	5.66E-07
		9.94E-06			3.98E-00			9.97E-06	
f_4 : Ackley	9.55E-06	8.80E-06	2.72E-07	9.65E-06	9.20E-06	2.00E-07	9.71E-06	8.93E-06	2.54E-07
		9.90E-06			9.97E-06			9.96E-06	
f_5 : Griewank	9.33E-06	7.94E-06	5.62E-07	1.28E-03	7.48E-06	2.97E-03	6.08E-03	8.78E-06	1.60E-02
		9.98E-06			9.86E-03			7.39E-02	
f_6 : Rastrigin	9.21E-06	7.85E-06	5.53E-07	9.27E-06	8.22E-06	4.09E-07	9.47E-06	8.84E-06	4.29E-07
		9.90E-06			9.94E-06			9.94E-06	
f_7 : Schwefel	-20718.37	-20949.14	157.50	-20426.20	-20795.39	205.61	-20291.54	-20801.77	320.16
		-20446.32			-20139.80			-19764.76	

Table 2. Results of DE, CDEDM and CDE for 100-dimensional benchmark problems

Method / Function	CDE			CDEDM			DE		
	f_{av}	f_b / f_w	σ_f	f_{av}	f_b / f_w	σ_f	f_{av}	f_b / f_w	σ_f
f_1 : Sphere	9.33E-06	7.41E-06	4.65E-07	9.54E-06	7.41E-06	5.30E-07	9.59E-06	8.33E-06	4.66E-07
		9.89E-06			9.88E-06			9.99E-06	
f_2 : Ellipsoid	9.13E-06	7.92E-06	3.88E-07	9.23E-06	8.42E-06	6.65E-07	5.89E-04	1.13E-04	3.05E-04
		9.95E-06			9.96E-06			1.36E-03	
f_3 : Rosenbrock	9.01E-06	6.81E-06	8.53E-07	2.57E-01	8.10E-06	8.78E-01	1.99E-01	8.11E-06	8.69E-01
		9.87E-06			4.01E-00			3.99E+00	
f_4 : Ackley	9.31E-06	7.90E-06	3.99E-07	9.36E-06	8.53E-06	4.11E-07	9.55E-06	8.94E-06	3.80E-07
		9.95E-06			9.90E-06			9.98E-06	
f_5 : Griewank	8.96E-06	6.56E-06	5.85E-07	9.10E-06	7.36E-06	1.06E-06	9.56E-04	8.45E-06	2.69E-03
		9.95E-06			9.95E-06			8.40E-03	
f_6 : Rastrigin	9.20E-06	6.83E-06	6.00E-07	9.33E-06	6.84E-06	6.68E-07	9.43E-06	8.13E-06	7.26E-07
		9.99E-06			9.98E-06			9.98E-06	
f_7 : Schwefel	-40493.87	-40951.78	441.63	-39531.39	-40822.13	647.32	-38869.76	-39726.06	708.83
		-39726.91			-38360.67			-37535.81	

N_p is the population size.

The average distance $d_{pa}(\%)$ is defined as follows:

$$d_{pa}(\%) = \frac{\overline{d_{pa}}}{L} \times 100\% \tag{21}$$

where $\overline{d_{pa}} = \frac{\sum_{i=1}^{N_p} \left\| X_i - X^* \right\|}{N_p}$ is the average distance between the population and the global optimum solution X^*.

The mutation strategies are:

In each generation, CDEDM and CDE randomly choose strategy from the strategy can including (1)DE/best/1，(2) DE/rand-to-best/1 ， (3)DE/rand/2; DE uses the mutation strategy DE/best/1.

The Parameters are:

A. Dimension n : 100

B. Number of populations N_p : 800 ($N_p = 5 \sim 10 \times n$) [17]

C. Maximal number of generations MaxGen：6000

D. Mutation factor F = 0.5~0.6

E. Crossover rate CR = 0.3 [9]

The parameters of CDE are:

A. Evaluated number of generations : 200

B. Number of better individuals : 400

C. Broaden ratio of the promising space : 10

The results of the above seven 100-dimensional benchmark functions are shown in Figs 4-24. Comparisons of the population convergence and the performance of d_{pg} and d_{pa} among the three algorithms are conducted for the uni-modal and the multi-modal functions separately.

(1) Uni-modal problems:

Although CDE performs worse than DE and CDEDM do in the beginning, CDEDM quickly converges to a better solution later in the search process. As shown in Figs. 10-12, although all the three algorithms take a long time to search for better

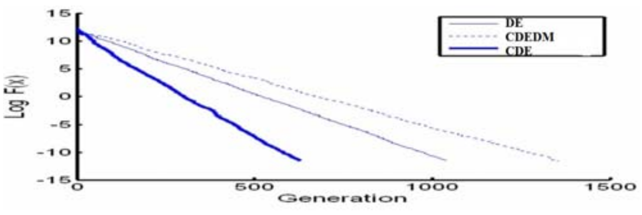

Fig. 4. Convergence process of the objective function value of f_1.

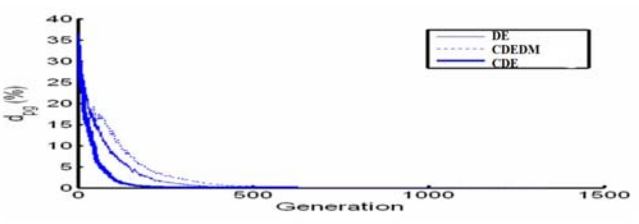

Fig. 5. The d_{pg} *history* of function f_1.

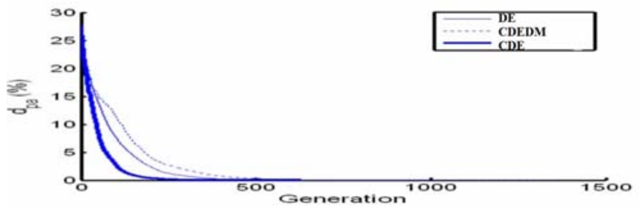

Fig. 6. The d_{pa} *history* of function f_1..

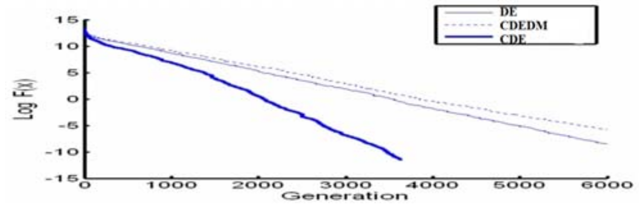

Fig. 7. Convergence process of the objective function value of f_2..

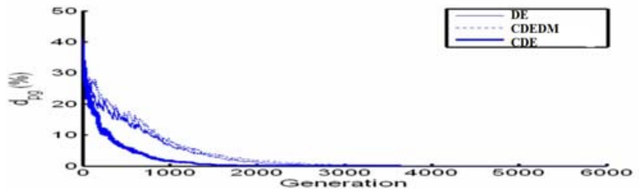

Fig. 8. The d_{pg} *history* of function f_2..

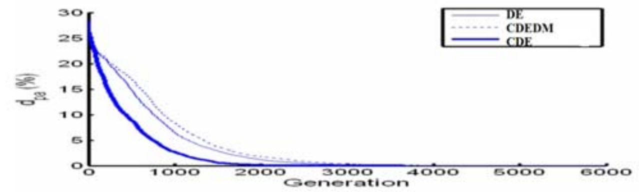

Fig. 9. The d_{pa} *history* of function f_2.

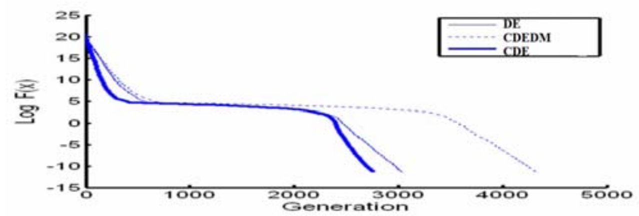

Fig. 10. Convergence process of the objective function value of f_3.

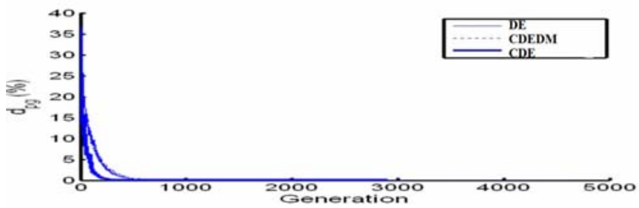

Fig. 11. The d_{pg} *history* of function f_3.

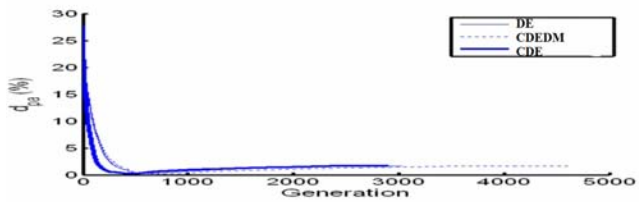

Fig. 12. The d_{pa} *history* of function f_3.

solution in the narrow region of function f_3, CDE is the quickest. Therefore, the mutation-garbage can strategy of CDE creates diversified searches from the beginning. With the grouping search strategy, it can also quickly lead the population to a better solution.

(2) Multi-modal problems:

As shown in Figs. 13-24, after incorporating the random selection strategy, CDEDM cannot lead population to the better solution, even though it has a diversified search capability. In contrast, CDE has both the sufficient diversity capacity of CDEDM and a better orientation for population. CDE can detect good solution in the search space in the beginning of the search and has the potential to find the global optimum. On functions f_4 and f_5, DE has similar convergence with CDE and can lead the population to move to good fitness spaces. As for functions f_6 and f_7, CDE has the best convergence performance, followed by DE. The population shift history diagrams of d_{pg} and d_{pa} show that the CDE's performance is the most prominent. CDE quickly leads the population to the vicinity of the global optimum. Finally, on function f_7 , the population shift history of d_{pg} in Fig. 23 shows that the population's diversity of CDEDM is strong, but it cannot find the best solution further. DE and

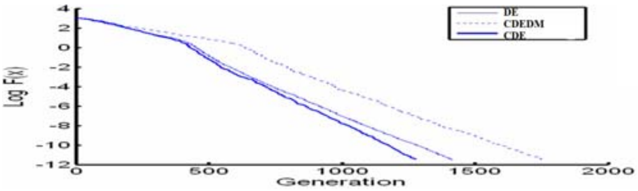

Fig. 13. Convergence process of the objective function value of f_4.

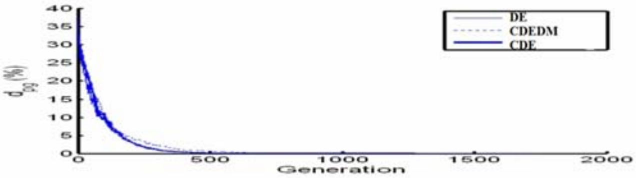

Fig. 14. The d_{pg} *history* of function f_4.

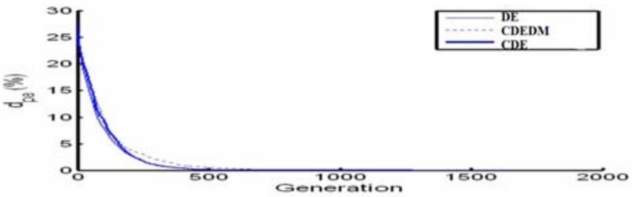

Fig. 15. The d_{pa} *history* of function f_4.

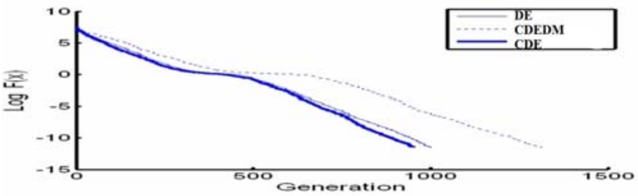

Fig. 16. Convergence process of the objective function value of f_5.

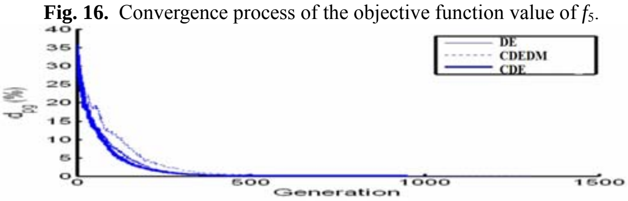

Fig. 17. The d_{pg} *history* of function f_5.

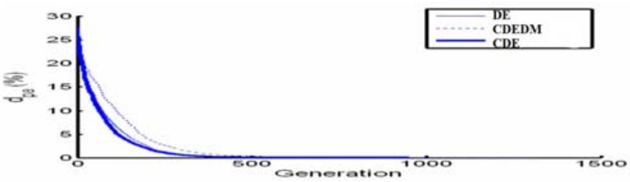

Fig. 18. The d_{pa} *history* of function f_5.

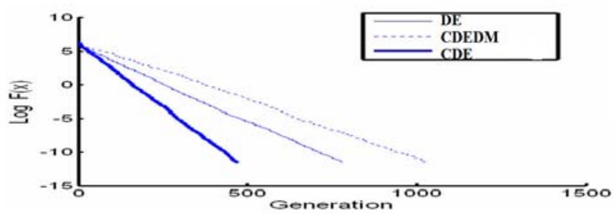

Fig. 19. Convergence process of the objective function value of f_6.

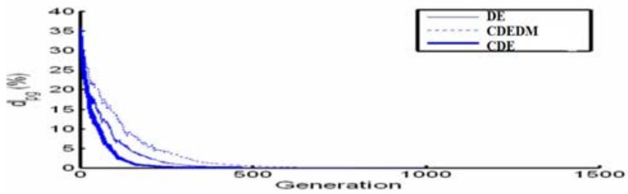

Fig. 20. The d_{pq} *history* of function f_6.

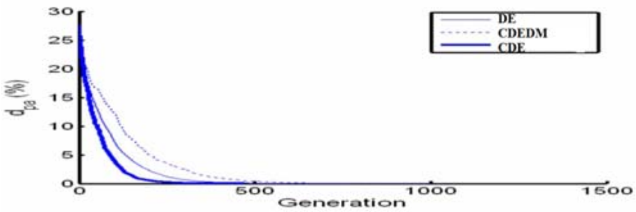

Fig. 21. The d_{pa} *history* of function f_6.

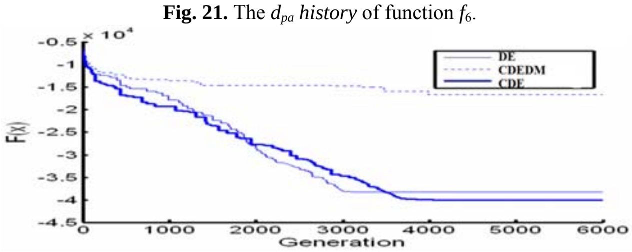

Fig. 22. Convergence process of the objective function value of f_7.

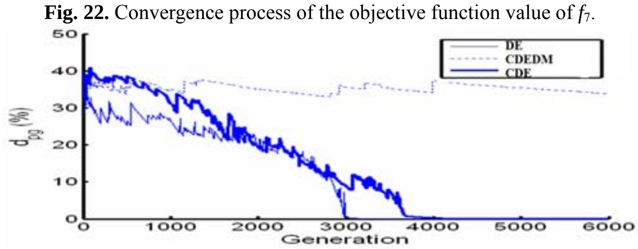

Fig. 23. The d_{pg} *history* of function f_7.

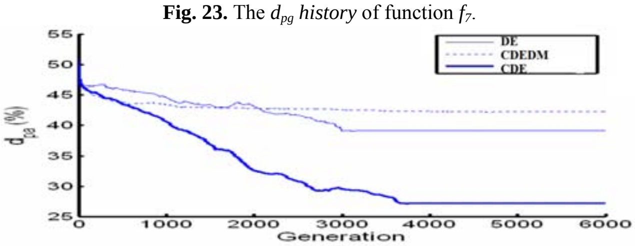

Fig. 24. The d_{pa} *history* of function f_7.

CDE have similar search trends in Figs. 23-24. The history diagrams of d_{pg} and d_{pa} indicate that CDE has the high diversity of population at the onset of searching, the intensification increasing over generations, and a better convergence achieved than the other two.

The experimental results of the above seven benchmark functions show that CDE's performance in exploration and exploitation is the best among three variants of DE and also validate that this paper has improved the performance of DE by introducing the garbage can decision model with group meeting into its search process.

VI. Conclusion

A new variant of differential evolution algorithm is developed by extending the thinking of the garbage can model with grouping meeting strategy, namely creative differential evolution algorithm (CDE). In the evolutionary process of this new variant DE, we propose mutation strategy can with several mutation strategies in the mutation step, mimicking open innovation decision model, and subpopulations search strategy mimicking integrate innovation decision model in process innovation decision model.

The experimental results on the seven benchmark functions with 50 and 100 dimensions have shown that CDE performs much better than DE and CDEDM in searching for the global optimum. The results of convergence process of objective function, dpg and dpa histories indicate that both good diversity and fast evolution speed can be obtained in CDE. Meanwhile, CDE saves the time spent on picking the suitable parameters and mutation strategy for optimization problems. The results show that CDE, which introduces the garbage can model with group meeting into DE, will not only enhance its search performance but also increase its ease of use.

Acknowledgement

We are grateful to National Science of Council (Republic of China) for the financial support to the study with a grant number of NSC 100-2221-E-019-014.

References

1. Abbass, H. A., "An evolutionary artificial neural networks approach for breast cancer diagnosis," Artificial Intelligence in Medicine, Vol. 25, No. 3, pp. 265-281 (2002).
2. Ali, M. M. and Törn, A., "Population set-based global optimization algorithm: Some modifications and numerical studies," Computers & Operation Research, Vol. 31, No. 10, pp. 1703–1725 (2004).
3. Cheng, S. L. and Hwang, C., "Optimal approximation of linear systems by a differential evolution algorithm," IEEE Transactions on Systems, Man and Cybernetics, Part A, Vol. 31, No. 6, pp. 698-707 (2001).
4. Cheong, F. and Lai, R., "Designing a hierarchical fuzzy logic controller using the differential evolution approach," Applied Soft Computing, Vol. 7, No. 2, pp. 481-491 (2007).
5. Chen,Y., Yang, S. and Nie, Z., "The Application of a Modified Differential Evolution Strategy to Some Array Pattern Synthesis Problems," IEEE Transactions On Antennas and Propagation, Vol.56, No. 7, pp. 1919-1927 (2008).
6. Cohen, M. D., March J. G. and Olsen, J. P., "A Garbage Can Model of Organizational Choice," Administrative Science Quarterly, Vol. 17, No. 1, pp. 1-25 (1972).
7. Dorigo, M., Maniezzo, V. and Colorni, A., "Ant System: Optimization by a Colony of Co-operating Agents," IEEE Transaction on System, Man, and Cybernetic-Part B: Cybernetic, Vol. 26, No. 1, pp. 29-41 (1996).
8. Feokitistov, V. and Janaqi, S., "Generalization of the Strategies in Differential Evolution," In: the Proceedings of the 18th International Parallel and Distributed processing Symposium, Vol. 7, pp. 165-170 (2004).
9. Gämperle, R. S., Müller, D. and Koumoutsakos, P., "A Parameter Study for Differential Evolution," International Conference on Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation, pp. 293-298 (2002).
10. Holland, J.H., Adaption in Natural and Artificial Systems, The University of Michigan Press (1975).
11. Kennedy, J. and Eberhart, R. C., "Particle Swarm Optimization," Proc. IEEE International Conference on Neural Networks, Vol. 4, pp. 1942-1948 (1995).
12. Kaelo , P. and Ali. M.M., "A numerical study of some modified differential evolution algorithms," European Journal of Operational Research, Vol. 169, No. 3, pp. 1176–1184 (2006).
13. Liu, W. B. and Wang, X. J., "An Evolutionary Game based Particle Swarm Optimization Algorithm," Journal of Computational and Applied Mathematics, Vol. 214, No. 1, pp. 30-35 (2008).
14. Lindblom, C. E., "The Science of "Muddling Through"," Public Administration Review, Vol.19, No. 2, pp. 79-88 (1959).
15. Nelder, J. A. and Mead, R., "A Simplex Method for Function Minimization," Computer Journal, Vol. 7, No. 4, pp. 308-313 (1965).
16. Nobakhti , A. and Wang, H., "A simple self-adaptive Differential Evolution algorithm with application on the ALSTOM gasifier," Applied Soft Computing, Vol. 8, No. 1, pp. 350–370 (2008).
17. Noman, N. and Iba, H., "Accelerating Differential Evolution Using an Adaptive Local Search," IEEE Transactions on Evol. Comput, Vol. 12, No. 1, pp. 107–125 (2008).
18. Qin ,A. K. and Suganthan, P. N., "Self-adaptive differential evolution algorithm for numerical optimization," in: The 2005 IEEE Congress on Evolutionary Computation CEC2005, Vol. 2, pp. 1785–1791 (2005).
19. Storn, R. and Price, K., "Differential Evolution- A Simple and Efficient Heuristic for Global Optimization Over Continuous Spaces," Journal of Global Optimization, Vol. 11, No. 4, pp.341-359 (1997).
20. Sun, J., Zhang, Q. and Tsang, E., "DE/EDA: A New Evolutionary Algorithm for Global Optimisation," Information Sciences, Vol. 169, No. 3-4, pp. 249-262 (2005).
21. Smith, J. M., Evolution and Theory of Game, Cambridge University Press, New York (1982).
22. Simon, H. A., "Some further notes on a class of skew distribution functions," Information and Control, Vol. 3, No. 1, pp. 80-88 (1960).
23. Simon, H. A., "Herbert. How Big is a Chunk?," Science, Vol.183, No.4124, pp. 482-488 (1974).
24. Teo, J., "Exploring dynamic self-adaptive populations in differential evolution," Soft computing—A fusion of foundations, methodologies and applications,Vol. 10, No. 8, pp. 673-684 (2006).
25. Taylor, F.W., Scientific Management: Comprising Shop Management, the Principles of Scientific Management Testimony before the Special House Committee, Harper & Row, New York (1947).
26. Wang, Y. J., Zhang, J. S. and Zhang, G. Y., "A dynamic clustering based differential evolution algorithm for global optimization," European Journal of Operational Research, Vol. 183, No. 1, pp. 56–73 (2007).
27. Wiegand, R. P., Liles, W. C. and De Jong, K. A., "Analysis Cooperative Co-evolution with Evolutionary Game Theory," Evol. Comput, Vol. 2, pp. 1600-1605 (2002).
28. Xu, R., Venayagamoorthy, G. K. and Wunsch D. C. II, "Modeling of gene regulatory networks with hybrid differential evolution and particle swarm optimization," Neural Networks, Vol. 20, No. 8, pp. 917–927. (2007).
29. Yang, G., Dong, Z. Y. and Wong, K. P., "A Modified Differential Evolution Algorithm With Fitness Sharing for Power System Planning," IEEE Transaction on Power Systems, Vol. 23, No. 2, pp. 514-522 (2008).
30. Yao, X., Liu, Y. and Lin, G., "Evolutionary programming made faster," IEEE Transactions on Evolutionary Computation, Vol. 3, No. 2, pp. 82–102 (1999).
31. Zaharie, D., "Control of Population Diversity and Adaptation in Differential Evolution Algorithms," In: Matousek, R., Osmera, P. (Eds.). In: Proceedings of the 9th International Conference on Soft Computing, Brno, pp.41-46 (2003).