# Adaptive Plan system using Differential Evolution with Genetic Algorithm

Hieu Pham
and Tam Bui
Functional Control Systems
Graduate School of Engineering and Science
Shibaura Institute of Technology
Japan
Email: pnh112@gmail.com

Hiroshi Hasegawa
College of Systems Engineering and Science
Shibaura Institute of Technology
Japan
Email: h-hase@shibaura-it.ac.jp
Telephone: +81-48-687-5179
Fax: +81-48-687-5197

***Abstract*—This paper describes a new proposed strategy of Adaptive Plan System using Differential Evolution (DE) with Genetic Algorithm (GA) called APGA/DE to solve large scale optimization problems, to reduce a large amount of calculation cost, and to improve stability in convergence to an optimal solution. This is an approach that combines the global search ability of GA and Adaptive Plan (AP) for local search ability. The proposed strategy incorporates new concept of AP using DE for Adaptive System (AS) with GA. The APGA/DE is applied to several benchmark functions with multi-dimensions to evaluate its performance. It is shown to be statistically significantly superior to other Evolutionary Algorithms (EAs), and Memetic Algorithms (MAs). We confirmed satisfactory performance through various benchmark tests.**

## I. Introduction

Evolutionary Algorithms (EAs) have been developed for solving combinatorial and numeric optimization problems. The most popular EA, Genetic Algorithm (GA) [1], [2], has been applied to various multimodal optimization problems with multi-dimensions. The validity of this method has been reported by many researchers. However, it requires a huge computational cost to obtain stability in the convergence to an optimal solution. To reduce the cost and to improve stability, a strategy that combines global and local search methods becomes necessary. As for this strategy, current research has proposed various methods. For instance, Memetic Algorithms (MAs) [3]–[7] are a class of stochastic global search heuristics in which EAs-based approaches are combined with local search techniques to improve the quality of the solutions created by evolution. MAs have proven very successful across the search ability for multimodal functions with multi-dimensions [8]. These methodologies need to choose suitably a best local search method combining with a global search method within the optimization process. Furthermore, since genetic operators are employed for a global search method within these algorithms, design variable vectors (DVs) which are renewed via a local search are encoded into its genes many times at its GA process. These certainly have the potential to break its improved chromosomes via gene manipulation by GA operators, even if these approaches choose a proper survival strategy.

To solve these problems and maintain the stability of the convergence to an optimal solution for multi-peak optimization problems with multiple dimensions, Hasegawa et al. proposed a new evolutionary algorithm called an Adaptive Plan system with Genetic Algorithm (APGA) [9].

A new evolutionary algorithm known as Differential Evolutionary (DE) was recently introduced and has garnered significant attention in the research literature with improved performance [10]–[12]. DE operates through similar computational steps as employed by a standard EA. However, compared with other forms of EAs [13], it hardly requires any parameter tuning and is very efficient and reliable. DE has many advantages including simplicity of implementation, robust, and is generally considered as an effetive global optimization algorithm. Neverthless, DE also has shortcomings as all other intelligent techniques such as local search ability, premature convergence, stagnation problems, control parameters, etc. Therefore, many researchers have done several attempts to overcome these problems and to improve the performance of the DE algorithm [14]–[19].

In this paper, we purposed a new strategy of Adaptive Plan System using DE with GA to solve large scale optimization problems, to reduce a large amount of calculation cost, and to improve the convergence to the optimal solution called APGA/DE.

The remainder of this paper is organized in the following manner. The basic concepts of DE is described in Section 2, Section 3 explains the algorithm of new proposed strategy (APGA/DE), and Section 4 discusses about the convergence to the optimal solution of multimodal benchmark functions. Finally, Section 5 includes some brief conclusions.

## II. Differential Evolution

Differential Evolution (DE) is an EA proposed by Storn and Price [10], also a population-based heuristic algorithm, which is simple to implement, requires little or no parameter tuning and is known for its remarkable performance for combinatorial optimization. DE is similar to other EAs particularly GA in the sense that it uses the same evolutionary operators such as selection, recombination, and mutation. However the

978-1-4673-4569-9/13/$31.00 ©2013 IEEE 

significant difference is that DE uses distance and direction information from the current population to guide the search process. The performance of DE depends on the manipulation of target vector and difference vector in order to obtain a trial vector.

### A. Mutation Operator

Mutation becomes the main operator in DE. For a $D$-dimensional search space, each target vector $X_{i,G}$, the most useful strategies of a mutant vector are

DE/rand/1

$$V_{i,G} = X_{r_1,G} + F \cdot (X_{r_2,G} - X_{r_3,G}) \ , \tag{1}$$

DE/best/1

$$V_{i,G} = X_{best,G} + F \cdot (X_{r_2,G} - X_{r_3,G}) \ , \tag{2}$$

DE/target-to-best/1

$$\begin{aligned} V_{i,G} = X_{i,G} &+ F \cdot (X_{best,G} - X_{i,G}) \\ &+ F \cdot (X_{r_2,G} - X_{r_3,G}) \ , \end{aligned} \tag{3}$$

DE/best/2

$$\begin{aligned} V_{i,G} = X_{best,G} &+ F \cdot (X_{r_1,G} - X_{r_2,G}) \\ &+ F \cdot (X_{r_3,G} - X_{r_4,G}) \ , \end{aligned} \tag{4}$$

DE/rand/2

$$\begin{aligned} V_{i,G} = X_{r_1,G} &+ F \cdot (X_{r_2,G} - X_{r_3,G}) \\ &+ F \cdot (X_{r_4,G} - X_{r_5,G}) \ , \end{aligned} \tag{5}$$

where $r_1,\ r_2,\ r_3,\ r_4,\ r_5 \in [1, 2, \ldots, NP]$ are mutually exclusive randomly chosen integers with a initiated population of $NP$, and all are different from the base index $i$. $G$ denotes subsequent generations, and $F > 0$ is a scaling factor which controls the amplification of differential evolution. $X_{best,G}$ is the best individual vector with the best fitness (lowest objective function value for a minimization) in the population.

### B. Crossover Operation

To enhance the potential diversity of the population, a crossover operation is introduced. The donor vector exchanges its components with the target vector to form the trial vector

$$U_{ij,G+1} = \begin{cases} V_{ij,G+1}, (rand_j \leq CR) \ or \ (j = j_{rand}) \\ X_{ij,G+1}, (rand_j \geq CR) \ and \ (j \neq j_{rand}) \end{cases} , \tag{6}$$

where $j = [1, 2, \ldots, D]$; $rand_j \in [0.0, 1.0]$; $CR$ is the crossover probability takes value in the range [0.0,1.0], and $j_{rand} \in [1, 2, \ldots, D]$ is the randomly chosen index.

### C. Selection

Selection is performed to determine whether the target vector or the trial vector survives to the next generation. The selection operation is described as

$$X_{i,G+1} = \begin{cases} U_{i,G}, & f(U_{i,G}) \leq f(X_{i,G}) \\ X_{i,G}, & f(U_{i,G}) > f(X_{i,G}) \end{cases} . \tag{7}$$

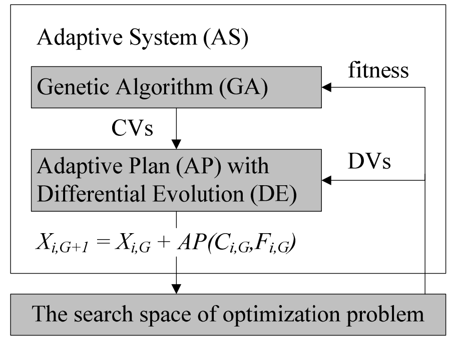


Fig. 1. APGA/DE Conceptual Strategy

## III. New Evolution Strategy: APGA/DE

With a view to global search, we proposed the new algorithm of Adaptive Plan system using DE with GA (APGA/DE). The proposed APGA/DE aims at incorporating new concept of AP to adjust into adaptive system of APGA using alternative operator of DE scheme.

### A. APGA/DE Algorithm

The APGA/DE generates a new candidate solution with adaptive system of APGA using the binomial crossover operation (6), and the conditions given by (7) of DE. The flow-chart of APGA/DE algorithm is shown in Fig. 1.

**Algorithm 1** The APGA/DE Pseudocode

```
Initialize population with CVs;
Generate initial DVs;
Evaluate individuals with initial DVs;
while (Termination Condition) do
    Generate DVs via AP with DE;
    Evaluate individuals with DVs;
    Select parents;
    Recombine to produce offspring for CVs;
    Mutate offspring for CVs;
    if (Restructuring Condition) then
        Restructure chromosome of offspring for CVs;
    end if
end while
```

### B. Adaptive Plan (AP)

Adaptive Plan with Genetic Algorithm (APGA) [9] that combines the global search ability of a GA and an Adaptive Plan with excellent local search ability is superior to other EAs, MAs [8]. The APGA concept differs in handling Design variable vectors (DVs) from general EAs based on GAs. EAs generally encode DVs into the genes of a chromosome, and handle them through GA operators. However, APGA completely separates DVs of global search and local search methods. It encodes Control variable vectors (CVs) of AP into its genes on Adaptive system (AS). Moreover, this separation strategy for DVs and chromosomes can solve MA problem of breaking chromosomes. The control variable vectors (CVs) steer the behavior of adaptive plan (AP) for a global search, and are renewed via genetic operations by estimating fitness value. For a local search, AP with DE generates next values of DVs by using CVs, scaling factor $F$ and current values of DVs according to the formula

$$X_{i,G+1} = X_{i,G} + AP(C_{i,G}, F_{i,G}) \ , \tag{8}$$

where $AP()$, $X$, $C$, $F$, and $G$ denote a function of AP, DVs, CVs, scaling factor of DE scheme and subsequent generations, respectively.

It is necessary that the AP realizes a local search process by applying various heuristics rules. In this paper, the plan introduces a DV generation formula using the alternative scheme of DE/target-to-best/1 (3) as in

$$\begin{aligned} AP(C_{i,G}, F_{i,G}) = F'_{i,G} \cdot [(X_{best,G} - X_{i,G}) \\ + (X_{k,G} - X_{q,G})] \ , \end{aligned} \tag{9}$$

$$F'_{i,G} = SP \cdot F_{i,G} \ , \tag{10}$$

$$SP = 2 \cdot C_{i,G} - 1 \ , \tag{11}$$

where $F'_{i,G}$ is given as a new control parameter. $SP$ and $F_{i,G} \in [0.0, 1.0]$ denote step size and scaling factor, respectively. $X_{best,G}$ is the best individual with the best fitness. $k$, $q$ are randomly chosen index from population, and $k \neq q \neq i$.

A step size $SP$ is defined by CVs for controlling a global behavior to prevent it falling into the local optimum. $C = [c_{i,j}, \ \ldots, \ c_{i,p}]$; $(0.0 \leq c_{i,j} \leq 1.0)$ is used so that it can change the direction to improve or worsen the objective function, and $C$ is encoded into a chromosome by 10 bit strings (shown in Fig. 2). In addition, $i$, $j$ and $p$ are the individual number, design variable number and its size, respectively.

### C. Self-adaptive Control Parameters

Recently, Brest et al. [15] proposed a self-adaption scheme for the DE control parameters. The scaling factor $F$ and crossover probability $CR$ are encoded into the individual and adjusted by introducing new parameters $\tau_1$, $\tau_2$. In this paper, we computed the new control parameters for next generation as

$$F_{i,G+1} = \begin{cases} F_l + rand_1 * F_u & with\ probability\ \tau_1 \\ F_{i,G} & else \end{cases} \ , \tag{12}$$

$$CR_{i,G+1} = \begin{cases} rand & with\ probability\ \tau_2 \\ CR_{i,G} & else \end{cases} \ , \tag{13}$$

where $F_l$ and $F_u$ are the lower and upper limits of $F$, $F_l = 0.1$ and $F_u = 0.9$, and $\tau_1 = \tau_2 = 0.1$. The new $F$ and $CR$, obtained before the mutation is performed, take value from [0.0,1.0].

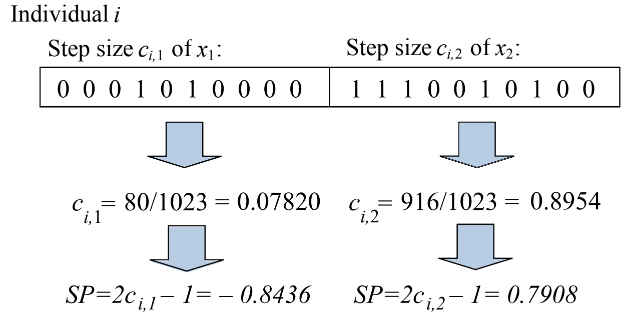


Fig. 2. Step size that defined by CVs for controlling a global behavior to prevent it falling into the local optimum

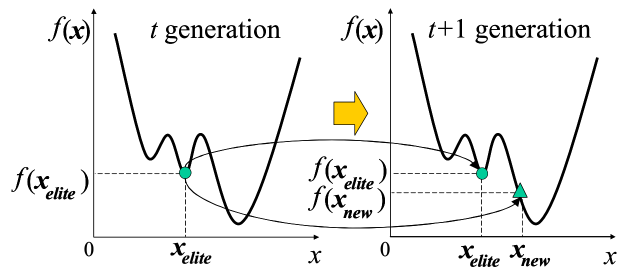


Fig. 3. Elite strategy, where the best individual survives in the next generation, is adopted during each generation process

### D. GA Operators

*1) Selection:* Selection is performed using a tournament size of 2 strategy to maintain the diverseness of individuals with a goal of keeping off an early convergence.

*2) Elite Strategy:* An elite strategy, where the best individual survives in the next generation, is adopted during each generation process. It is necessary to assume that the best individual, i.e., as for the elite individual, generates two behaviors of AP by updating DVs with AP, not GA operators. Therefore, its strategy replicates the best individual to two elite individuals, and keeps them to next generation. As shown in Fig. 3, DVs of one of them ($\Delta$ symbol) is renewed by AP, and its CVs which are coded into chromosome arent changed by GA operators. Another one ($\circ$ symbol) is that both DVs and CVs are not renewed, and are kept to next generation as an elite individual at the same search point.

*3) Crossover and Mutation:* In order to pick up the best values of each CV, a single point crossover is used for the string of each CV. This can be considered to be a uniform crossover for the string of the chromosome. Mutation are performed for each string at mutation ratio on each generation, and set to maintain the strings diverse.

*4) Recombination of Genes:* At following conditions, the genetic information on chromosome of individual is recombined by uniform random function

- One fitness value occupies 80% of the fitness of all individuals.
- One chromosome occupies 80% of the population.

TABLE I
PARAMETER SETTINGS FOR BENCHMARK TESTS

| Operator | Control Parameter | Set value |
|---|---|---|
| DE | Initial scaling factor | $F = 0.1$ |
| | Initial crossover | $CR = 0.5$ |
| | Limits of scaling factor | $F_l = 0.1$; $F_u = 0.9$ |
| | Probability | $\tau_1 = \tau_2 = 0.1$ |
| GA | Selection | 1.0 |
| | Crossover | 0.8 |
| | Mutation | 0.1 |

The population size: 100; The terminal generation: 1500

## IV. NUMERICAL EXPERIMENTS

The numerical experiments are performed 50 independent trials for each function. The parameter settings used in solving the benchmark functions are given in Table I. The initial seed number is randomly varied during every trial. The lower and upper limits of scaling factor $F$, $F_l = 0.1$ and $F_u = 0.9$, the initial control parameters $F$ and $CR$ are respectively set by 0.1 and 0.5 for the performance of DE. The GA parameters, selection ratio, crossover ratio and mutation ratio are 1.0, 0.8 and 0.1, respectively. The population size is 100 individuals and the terminal generation is 1500 generations.

### *A. Benchmark Functions*

For the APGA/DE, we estimated the stability of the convergence to the optimal solution by using five benchmark functions with 30, and 100 dimensions: Rastrigin (RA), Griewank (GR), Ridge (RI), Ackley (AC), and Rosenbrock (RO). These functions are given as follows

$$RA : f_1 = 10D + \sum_{i=1}^{D}[x_i^2 - 10cos(2\pi x_i)] \ , \qquad (14)$$

$$RI : f_2 = \sum_{i=1}^{D}\left(\sum_{j=1}^{i} x_j\right)^2 \ , \qquad (15)$$

$$GR : f_3 = 1 + \sum_{i=1}^{D}\frac{x_i^2}{4000} - \prod_{i=1}^{D} cos\left(\frac{x_i}{\sqrt{i}}\right) \ , \qquad (16)$$

$$AC : f_4 = -20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right) + 20 + e \ , \qquad (17)$$

$$RO : f_5 = \sum_{i=1}^{D}\left[100(x_{i+1} + 1 - (x_i+1)^2)^2 + x_i^2\right] \ . \qquad (18)$$

Table II lists their characteristics, including the terms epistasis, multimodal, and steepness. A more detailed description of each function is given in [20].

TABLE II
CHARACTERISTICS OF BENCHMARK FUNCTIONS

| Function | Epistasis | Multimodal | Steepness |
|---|---|---|---|
| RA | No | Yes | Average |
| RI | Yes | No | Average |
| GR | Yes | Yes | Small |
| AC | No | Yes | Average |
| RO | Yes | No | Big |

TABLE III
DESIGN RANGE AND OPTIMUM VALUE OF BENCHMARK FUNCTIONS

| Function | Design Range | Optimum Value |
|---|---|---|
| RA | $[-5.12, 5.12]^D$ | $f_1(0) = 0$ |
| RI | $[-100, 100]^D$ | $f_2(0) = 0$ |
| GR | $[-600, 600]^D$ | $f_3(0) = 0$ |
| AC | $[-32, 32]^D$ | $f_4(0) = 0$ |
| RO | $[-30, 30]^D$ | $f_5(0) = 0$ |

TABLE IV
AVERAGE RESULTS, OVER 50 TRIALS WITH 30 DIMENSIONS. "MEAN BEST" INDICATES AVERAGE OF OPTIMUM VALUES OBTAINED AND "STD DEV" STANDS FOR STANDARD DEVIATION

| Function | Gen. | NFE | Mean Best | Std Dev |
|---|---|---|---|---|
| RA | 113 | 11,300 | 0.00E+00 | 0.00E+00 |
| RI | 553 | 55,300 | 0.00E+00 | 0.00E+00 |
| GR | 101 | 10,100 | 0.00E+00 | 0.00E+00 |
| AC | 144 | 14,400 | 4.44E-16 | 0.00E+00 |
| RO | 355 | 35,500 | 0.00E+00 | 0.00E+00 |

TABLE V
AVERAGE RESULTS, OVER 50 TRIALS WITH 100 DIMENSIONS. "MEAN BEST" INDICATES AVERAGE OF OPTIMUM VALUES OBTAINED AND "STD DEV" STANDS FOR STANDARD DEVIATION

| Function | Gen. | NFE | Mean Best | Std Dev |
|---|---|---|---|---|
| RA | 133 | 13,300 | 0.00E+00 | 0.00E+00 |
| RI | 795 | 79,500 | 0.00E+00 | 0.00E+00 |
| GR | 129 | 12,900 | 0.00E+00 | 0.00E+00 |
| AC | 199 | 19,900 | 4.44E-16 | 0.00E+00 |
| RO | 478 | 47,800 | 0.00E+00 | 0.00E+00 |

Their design range and optimum value are summarized in Table III. All functions are minimized to zero ($ESP = 1.7e-308$), when optimal DVs $X = 0$ are obtained. If the search point attains an optimal solution or a current generation process reaches the termination generation, the search process is terminated.

### *B. Experiment Results*

The experiment results, average generations required to reach the global optimum of all benchmark functions by the APGA/DE are given in Table IV, and Table V. "Mean best" indicates average of optimum values obtained and "Std Dev" stands for standard deviation. The solution of all benchmark functions reach their global optimum solutions, and the success rate of optimal solution is 100%.

TABLE VI
Comparison of DE, jDE, ADE and APGA/DE Algorithm ($D = 30$, Population Size 100, and Max Generation 1500)

| Function | DE [10] Mean best (Std Dev) | jDE [15] Mean best (Std Dev) | ADE [19] Mean best (Std Dev) | APGA/DE Mean best (Std Dev) |
|---|---|---|---|---|
| RA | 173.405 (13.841) | 1.5E-15 (4.8E-15) | 0.0E+00 (1) (0.0E+00) | 0.00E+00 (0.00E+00) |
| RI | 1.630860 (0.886153) | 0.090075 (0.080178) | – | 0.00E+00 (0.00E+00) |
| GR | 2.9E-13 (4.2E-13) | 0 0 | 0.0E+00 (2) (0.0E+00) | 0.00E+00 (0.00E+00) |
| AC | 9.7E-08 (4.2E-08) | 7.7E-15 (1.4E-15) | 6.93E-11 (3.10E-11) | 4.44E-16 (0.00E+00) |
| RO | 7.8E-09 (5.8E-09) | 3.1E-15 (8.3E-15) | 3.75E-05 (3) (8.90E-05) | 0.00E+00 (0.00E+00) |

(1) Gen. No 5000; (2) Gen. No 2000; (3) Gen. No 3000

TABLE VII
Comparison of GA, PSO, ABC and APGA/DE Algorithm ($D = 30$, Population Size 125, and Max Generation 1000)

| Function | GA [21] Mean best (Std Dev) | PSO [21] Mean best (Std Dev) | ABC [22] Mean best (Std Dev) | APGA/DE Mean best (Std Dev) |
|---|---|---|---|---|
| RA | 10.4388 (2.6386) | 32.476 (6.9521) | 0.033874 (0.181557) | 0.00E+00 (0.00E+00) |
| RI | – | – | – | 0.00E+00 (0.00E+00) |
| GR | 1.2342 (0.11045) | 0.011151 (0.014209) | 2.87E-09 (8.45E-10) | 0.00E+00 (0.00E+00) |
| AC | 1.0989 (0.24956) | 1.49E-06 (1.86E-06) | 3.00E-12 (5.00E-12) | 4.44E-16 (0.00E+00) |
| RO | 166.283 (59.5102) | 402.54 (633.65) | 0.219626 (0.152742) | 0.00E+00 (0.00E+00) |

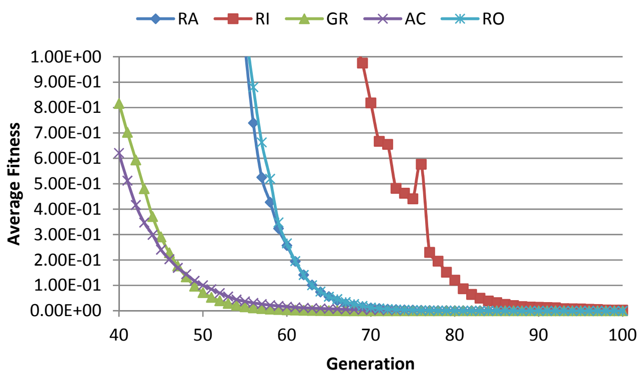


Fig. 4. Convergence, average fitnesses of all individuals by APGA/DE with 30 dimensions and population size 100

Next, Fig. 4 shows diagram for the convergence, average fitness values of all individuals in the population until the APGA/DE reaches the global optimum solutions with all benchmark functions, again to confirm above mentioned results.

As a result, its validity confirms that this strategy can dramatically reduce the computation cost and improve the stability of the convergence to the optimal solution more significantly.

### *C. Comparison for Robustness*

To show the effects of the APGA/DE, we compared to other EAs such as original DE [10], Self-adaptive DE (jDE) [15], Advanced DE (ADE) [19], GA [21], Particle Swarm Optimization (PSO) [21], and Artificial Bee Colony (ABC) [22] that used maximum number of generations and the population size as in [13], [21]. From the comparison as given in Table VI and Table VII, we confirmed that APGA/DE algorithm outperformed other techniques, and it converged the global optimal solution with a high probability.

In particular, it showed that the computation cost could be reduced dramatically, and the convergence towards the optimal solution could be improved more significantly.

Overall, the APGA/DE was capable of attaining robustness, high quality, low calculation cost, and efficient performance on many benchmark problems.

## V. Conclusion

In this paper, overcome the computational complexity, a new strategy of Adaptive Plan system using Differential Evolution with Genetic Algorithm called APGA/DE has been proposed to solve large scale optimization problems, to reduce a large amount of calculation cost, and to improve the convergence to the optimal solution. Then, we verified the effectiveness of APGA/DE algorithm by the numerical experiments performed five benchmark tests.

We confirmed that the APGA/DE reduces the calculation cost and dramatically improves the convergence towards the optimal solution. Moreover, it could solve large scale optimization problems with high probability.

About a solution of the problem of cost reduction, minimum time and maximum reliability, it is a future work.

Finally, this study plans to do a comparison with the sensitivity plan of the AP by applying other methods on constrained real-parameters and dynamic optimization problems.

## References


[1] J. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan 1975, MIT Press, 1992.

[2] D.E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison - Wesley, 1989.

[3] Y.S. Ong and A.J. Keane, *Meta-Lamarckian Learning in Memetic Algorithms*, IEEE Transactions on Evolutionary Computation, 8(2), 99–110, 2004.

[4] Y.S. Ong, M.H. Lim, N. Zhu and K.W. Wong, *Classification of Adaptive Memetic Algorithms: A Comparative Study*, IEEE Transactions on Systems, Man and Cybnetics Part B, 36(1), 141–152, 2006.

[5] F. Neri, V. Tirronen, T. Kärkkäinen and T. Rossi, *Fitness Diversity Based Adaptation in Multimeme Algorithms: A Comparative Study*, IEEE Congress on Evolutionary Computation, pp. 2374–2381, 2007.

[6] A. Caponio, G.L. Cascella, F. Neri, N. Salvatore and M. Sumner, *A Fast Adaptive Memetic Algorithm for Online and Offline Control Design of PMSM Drives*, IEEE Transactions on Systems, Man and Cybernetics Part B, Special Issue on Memetic Algorithms, 37(1), 28–41, 2007.

[7] V. Tirronen, F. Neri, T. Kärkkäinen, K. Majava and T. Rossi, *An Enhanced Memetic Differential Evolution in Filter Design for Defect Detection in Paper Production*, Evolutionary Computation Journal, MIT Press, 16(4), 529–555, 2008.
[8] J.E. Smith, W.E. Hart and N. Krasnogor, *Recent Advances in Memetic Algorithms* Springer, 2005.
[9] H. Hasegawa, *Adaptive Plan System with Genetic Algorithm based on Synthesis of Local and Global Search Method for Multi-peak Optimization Problems*, Proceedings of the 6th EUROSIM Congress on Modelling and Simulation, 2007.
[10] R. Storn and K. Price, *Differential Evolution - a simple and efficient Heuristic for global optimization over continuous spaces*, Journal Global Optimization, 11(4), 341–357, 1997.
[11] K. Price, R. Storn and J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Springer-Verlag, Berlin, 2005.
[12] S. Das and P.N. Suganthan, *Differential evolution - A survey of the State-of-the-Art*, IEEE Transactions on Evolutionary Computation, 15(1), 4–31, 2011.
[13] J. Vesterstroem and R. Thomsen, *A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems*, Proc. IEEE Congr. Evolutionary Computation, pp. 1980-1987, 2004.
[14] J. Liu and J. Lampinen, *A fuzzy adaptive differential evolution algorithm*, Soft Computing - A Fusion of Foundations, Methodologies and Applications, 9(6), 448-462, 2005.
[15] J. Brest, S. Greiner, B. Bošković, M. Mernik and V. Žumer, *Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems*, IEEE Trans. Evol. Comput., 10(6), 646–657, 2006.
[16] M. Omran, A.P. Engelbrecht and A. Salman, *Empirical analysis of self-adaptive differential evolution*, European Journal of Operations Research, 183(2), 785–804, 2007.
[17] N. Noman and H. Iba, *Accelerating differential evolution using an adaptive local Search*, IEEE Transactions on Evolutionary Computation, 12(1), 107–125, 2008.
[18] Y. Xu, L. Wang and L. Li, *An effective hybrid algorithm based on simplex search and differential evolution for global optimization*, In Proc. ICIC, pp. 341–350, 2009.
[19] A. Wagdy Mohamed, H.Z. Sabry and A. Farhat, *Advanced Differential Evolution algorithm for global numerical optimization*, IEEE International Conference on Computer Applications and Industrial Electronics (ICCAIE), pp. 156–161, 2011.
[20] X. Yao, Y. Liu and G. Lin, *Evolutionary programming made faster*, IEEE Trans. Evol. Comput., 3(2), 82–102, 1999.
[21] D. Srinivasan and T.H. Seow, *Evolutionary Computation*, IEEE Congress on Modelling and Simulation, 2010.
[22] D. Karaboga and B. Basturk, *A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm*, Journal Global Optimization, 39, 459–471, 2006.