A New Hybrid Differential Evolution with Wavelet Based Mutation and Crossover

J.C.Y. Lai[*], C.W. Yeung[†] and F.H.F. Leung[✦]
[*]Centre for Signal Processing, Dept. of Electronic and Information Engg., The Hong Kong Polytechnic University, Hung Ham, Hong Kong. E-mail: 08900438r@polyu.edu.hk
[†]Centre for Signal Processing, Dept. of Electronic and Information Engg., The Hong Kong Polytechnic University, Hung Ham, Hong Kong. E-mail: 07900256r@polyu.edu.hk
[✦]Centre for Signal Processing, Dept. of Electronic and Information Engg., The Hong Kong Polytechnic University, Hung Ham, Hong Kong. E-mail: enfrank@inet.polyu.edu.hk

***Abstract*— An improved Differential Evolution (DE) that incorporates wavelet-based mutation and crossover operations is proposed. In the mutation operation, the scaling factor is controlled by a wavelet function. In the crossover operation, the trial population vectors are modified by a wavelet function. The wavelet theory applied is to enhance DE in exploring the solution space more effectively for a better solution. A suite of benchmark test functions is employed to evaluate the performance of the proposed method. It is shown empirically that the proposed method outperforms significantly the conventional methods in terms of convergence speed, solution quality and solution stability.**

I. INTRODUCTION

Differential Evolution (DE) has been well accepted as a powerful algorithm for handling optimization problems during the last decade. Proposed by Storn and Price [1], DE is a population based stochastic optimization algorithm that searches the solution space by using the weighted difference between two population vectors to determine a third vector. No separate probability distribution has to be used so that the scheme is completely self-organizing [1] [12]. It is a new member to the class of Evolutionary Algorithms (EA) that imitate the process of biological evolution. Owing to the population based strategy, EAs are less possibly getting trapped in a locally optimal solution. As a result, many researchers view EAs as global optimization algorithms. Important examples of EAs include the Genetic Algorithm (GA) [5] and Evolutionary Programming (EP) [6].

Similar to GA, DE uses evolutionary operations to guide the population evolving towards the global solution within the given solution space. Comparing with other optimization algorithms, DE is easy to implement, requires fewer parameters for tuning, and have a relatively fast convergence speed. A simple vector subtraction is able to generate a random direction of exploration over the solution space. DE can also offer a high degree of variations for the population to search the solution space. It has been successfully applied in a wide range of optimization problems such as data clustering [2], power plant control [3], optimization of non-linear functions [4], etc. However, for maintaining the diversity from one generation of the population to the next, mutation takes an important role in the evolution process. The presence of mutation can help assuring the reached solution is a global optimum; but a too vigorous mutation in every iteration step may slow down or even destroy the convergence of the algorithm.

On doing the mutation and crossover operation, we can have the solution space to be more widely explored in the early stage of the search by setting a larger searching space; and it is more likely to obtain a fine-tuned global solution in the later stage of the search by setting a smaller searching space, based on the properties of wavelet [7]. The wavelet is a tool to model seismic signals by combining dilations and translations of a simple, oscillatory function (mother wavelet) of a finite duration [9]. Its properties enable us to improve the performance of DE. In this paper, mutation and crossover operations with a dynamic searching space by incorporating some wavelet functions [8] are proposed. The resulting mutation and crossover operations aid the DE to perform more efficiently and provide a faster convergence than the standard DE [1] in finding the solutions for a suite of benchmark test functions. In addition, it achieves better solution quality and higher solution stability.

This paper is organized as follows. Section II presents the operation of DE with wavelet mutation and wavelet crossover. Experimental study and analysis are given in Section III. Benchmark test functions are used to evaluate the performance of the proposed method. A conclusion will be drawn in Section IV.

II. DE WITH WAVELET MUTATION AND WAVELET CROSSOVER

To realize DE, a randomly generated population over the solution space will first be obtained. The population of solution vectors are then successively updated and swapped; until the population converge to the optimum. The pseudo code for the standard DE (SDE) process is shown in Fig. 1. In this paper, a DE with wavelet mutation and wavelet crossover (WMWC-DE) is proposed and the pseudo code for it is shown in Fig. 2. The details of both the SDE and the WMWC-DE are discussed as follows.

A. Standard Differential Evolution (SDE)

DE attempts to maintain a population of Np vectors for each generation of evolution, with each vector contains D elements of parameters. Let $P_{x,g}$ be the population of the current generation g, and $\mathbf{x}_{i,g}$ be the i-th vector in this population:

$$\begin{aligned} P_{x,g} &= (\mathbf{x}_{i,g}), i = 0,1,..., Np-1; g = 0,1,..., g_{\max} \\ \mathbf{x}_{i,g} &= (x_{j,i,g}), j = 0,1,..., D-1. \end{aligned} \quad (1)$$

Before the population can be initialized over the solution space, the boundary of the searching space should be specified. The population should be uniformly and randomly distributed in the searching space. Once initialized, DE creates a mutated vector, $\mathbf{v}_{i,g}$ for each target vector $\mathbf{x}_{i,g}$ by using the mutation operation. In particular, DE adds a scaled, randomly sampled, vector difference to form a third vector. The mutation operation is realized by the following equation:

$$\mathbf{v}_{i,g} = \mathbf{x}_{i,g} + F \cdot (\mathbf{x}_{r_1,g} - \mathbf{x}_{r_2,g}) \quad (2)$$

where F is the scaling factor; r_1 and r_2 are two different integers which are randomly generated from {0, 1, ..., Np–1}. The number of mutations taking place in each generation is also random. To complement the differential mutation search strategy and increase the diversity of the perturbed parameter vectors, DE employs a method called uniform crossover for the mutated vectors. Each vector element pair $x_{j,i,g}$ and $v_{j,i,g}$ generates a new trial vector element $u_{j,i,g}$. The crossover operation is realized by the following equation:

$$\mathbf{u}_{i,g} = (u_{j,i,g}) = \begin{cases} (v_{j,i,g}) & \text{if } (rand_j(0,1) \le Cr \\ (x_{j,i,g}) & \text{otherwise.} \end{cases} \quad (3)$$

where $Cr \in [0, 1]$ is called the crossover rate, which is a user-defined value that controls the fraction of parameters that are copied from the mutant. $rand_j(0,1)$ generates a random value between 0 and 1 for the j-th parameter. The algorithm also ensures $u_{j,i,g}$ gets at least one parameter value as $x_{j,i,g}$ [1]. Then the population is updated. If the trial vector has the fitness function value lower than that of the target vector, replace the target vector in the next generation; otherwise the target vector retains its place in the population for at least one generation of iteration. The selection operation is therefore realized by the following equation:

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g} & \text{if } f(\mathbf{u}_{i,g}) \le f(\mathbf{x}_{i,g}) \\ \mathbf{x}_{i,g} & \text{otherwise.} \end{cases} \quad (4)$$

where $f(\cdot)$ is the fitness function. Because of this selection operation, DE is expected to have high optimization ability. When the condition to stop further evolution is satisfied, for example, a preset maximum number of iteration has been reached, the algorithm ends with the best solution as the final solution (see Fig. 1).

```
begin
   Initialize the population
   While (not termination condition) do
      begin
         Mutation operation by equation (2)
         Crossover operation by equation (3)
         Evaluation of the function
         Select the best vector by equation (4)
      end
end
```

Fig. 1. Pseudo code for SDE.

```
begin
   Initialize the population
   While (not termination condition) do
      begin
         Update the new value of F by equation (14)
         Mutation operation by equation (2)
         Crossover operation by equation (3)
         Modifying the trial population vectors by
         equation (14)
         Evaluation of the function
         Select the best vector by equation (4)
      end
end
```

Fig. 2. Pseudo code for the proposed DE.

B. Differential Evolution with Wavelet Mutation and Wavelet Crossover (WMWC-DE)

In the SDE mutation operation, the value of F in (2) is a fixed value within the range of [0, 1] determined based on the kind of application. The choice of this value relies very much on experience or expert knowledge. Yet, a fixed value of F takes no advantage of the benefit brought by the evolution. We propose the value of F to diminish with the increase of the number of iteration. Moreover, for some complex optimization problems like finding the minimum point of a multimodal functions with many local minima, a large number of iteration for solving the problem is required in SDE. It reduces the efficiency of the SDE. This leads to the proposed WMWC-DE in which the value F is determined by a wavelet mutation function. The degree of freedom of the trial vector will then be increased. More ‘random’ vector directions would be generated during the mutation operation. Moreover, in the crossover operation, we proposed a second wavelet mutation that varies the searching space based on the wavelet theory. As the wavelet function output is inversely proportional to the number of iteration; when the searching population is approaching the optimal solution, the effect of the wavelet mutation and crossover operations will be decreasing until the DE ends eventually (see Fig. 2.) By adopting this method, the effort on searching and evaluating those local minima, which are far away from the global minimum, in the later iteration is reduced. Therefore the total number of iteration deceases. The result is a wavelet-mutation-wavelet-crossover-based DE (WMWC-DE).

C. Wavelet Mutation and Wavelet Crossover

1. Wavelet theory

Certain seismic signals can be modelled by combining translations and dilations of an oscillatory function with a finite duration called a "wavelet". A continuous function $\psi(x)$ is called a "mother wavelet" or "wavelet" if it satisfies the following properties:

Property 1:

$$\int_{-\infty}^{+\infty} \psi(x)dx = 0 \tag{5}$$

In other words, the total positive momentum of $\psi(x)$ is equal to the total negative momentum of $\psi(x)$.

Property 2:

$$\int_{-\infty}^{+\infty} |\psi(x)|^2 dx < \infty \tag{6}$$

Hence, most of the energy in $\psi(x)$ is confined to a finite duration and bounded. The Morlet wavelet [2], as shown in Fig. 3, is an example mother wavelet:

$$\psi(x) = e^{-x^2/2}\cos(5x) \tag{7}$$

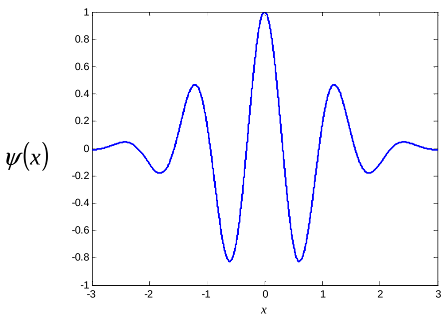

Fig. 3. Morlet wavelet.

The Morlet wavelet integrates to zero (*Property 1*). Over 99% of the total energy of the function is contained in the interval of $-2.5 \le x \le 2.5$ (*Property 2*). In order to control the magnitude and the position of $\psi(x)$, a function $\psi_{a,b}(x)$ is defined as follows.

$$\psi_{a,b}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x-b}{a}\right) \tag{8}$$

where a is the dilation parameter and b is the translation parameter. Notice that

$$\psi_{1,0}(x) = \psi(x) \tag{9}$$

$$\psi_{a,0}(x) = \frac{1}{\sqrt{a}}\psi\left(\frac{x}{a}\right). \tag{10}$$

It follows that $\psi_{a,0}(x)$ is an amplitude-scaled version of $\psi(x)$. Fig. 4 shows different dilations of the Morlet wavelet. The amplitude of $\psi_{a,0}(x)$ will be scaled down as the dilation parameter a increases. This property is used to do the mutation operation in order to enhance the searching performance.

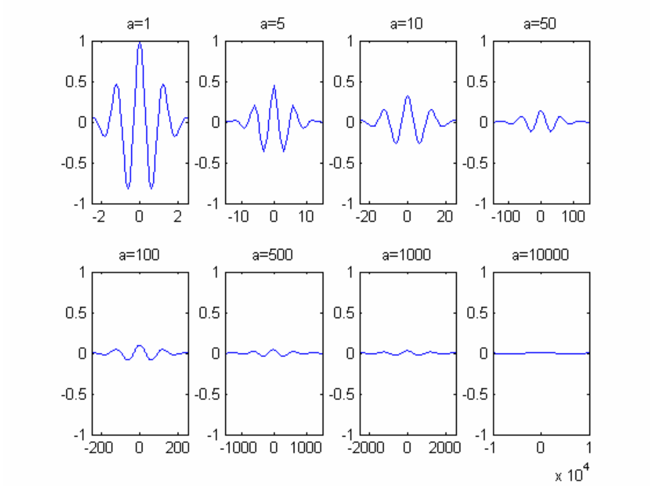

Fig. 4. Morlet wavelet dilated by different values of the parameter a (x-axis: a, y-axis: $\psi_{a,0}(x)$.)

2. Operation of wavelet mutation

The mutation operation is used to mutate the vectors in the population. The proposed wavelet mutation (WM) operation exhibits a fine-tuning ability. Consider (2), the mutation operation is modified as follows.

$$\mathbf{v}_{i,g} = \mathbf{x}_{r_0,g} + F \cdot \left(\left|\mathbf{x}_{r_1,g} - \mathbf{x}_{r_2,g}\right|\right), \tag{11}$$

where

$$F = \psi_{a,0}(\varphi), \tag{12}$$

$$F = \frac{1}{\sqrt{a}}\psi\left(\frac{\varphi}{a}\right) \tag{13}$$

By using the Morlet wavelet in (7) as the mother wavelet,

$$F = \frac{1}{\sqrt{a}} e^{-\left(\frac{\varphi}{a}\right)^2/2} \cos\left(5\left(\frac{\varphi}{a}\right)\right), \tag{14}$$

where

$$a = e^{-\ln(g)\times\left(1-\frac{t}{T}\right)^{\zeta_{wm}} + \ln(g)} \tag{15}$$

T is the total number of iteration and t is the current number of iteration; ζ_{wm} is the shape parameter of the monotonic increasing function, g is the upper limit of the parameter a. If F is positive approaching 1 or F is negative approaching –1, the mutation will tend to a maximum. Conversely, when F approaches 0, the mutation will tend to a minimum. A larger value of $|F|$ gives a larger searching space for the solution. When $|F|$ is small, it gives a smaller searching space for fine-tuning.

3. Operation of wavelet crossover

The crossover operation is done with respect to the elements of the trial vector (after mutation) in DE. In general, various methods like uniform crossover or non-uniform crossover [8, 10] can be employed to realize the crossover operation. The proposed wavelet crossover (WC) operation, which exhibits a fine-tuning ability, is realized by adding a second wavelet mutation following the original crossover operation. The details are as follows. The crossover after the first mutation takes place according to (3). Let $\mathbf{u}_{i,g} = \left(u_{0,i,g}, u_{1,i,g}, \ldots, u_{D-1,i,g}\right)$ (where g is the current generation number and D is the number of elements in the vector) be the i-th vector after crossover for the second wavelet mutation. Its element value $u_{j,i,g}$ is inside the vector element's boundary [$para_{\min}^j$, $para_{\max}^j$]. The resulting vector is given by $\bar{\mathbf{u}}_{i,g} = \left(\bar{u}_{0,i,g}, \bar{u}_{1,i,g}, \ldots, \bar{u}_{D-1,i,g}\right)$, and

$$\bar{u}_{j,i,g} = \begin{cases} u_{j,i,g} + \sigma\times\left(para_{\max}^j - u_{j,i,g}\right) \text{ if } \sigma > 0 \\ u_{j,i,g} + \sigma\times\left(u_{j,i,g} - para_{\min}^j\right) \text{ if } \sigma \le 0 \end{cases}, \tag{16}$$

$$\sigma = \psi_{a,0}(\varphi) \tag{17}$$

$$\sigma = \frac{1}{\sqrt{a}}\psi\left(\frac{\varphi}{a}\right) \tag{18}$$

By using the Morlet wavelet in (8) as the mother wavelet,

$$\sigma = \frac{1}{\sqrt{a}} e^{-\left(\frac{\varphi}{a}\right)^2/2} \cos\left(5\left(\frac{\varphi}{a}\right)\right) \tag{19}$$

If σ is positive ($\sigma > 0$) approaching 1, the mutation will tend to a maximum. Conversely, if σ is negative ($\sigma \le 0$) approaching –1, the mutation will tend to a minimum. A larger value of $|\sigma|$ gives a larger searching space for the solution.

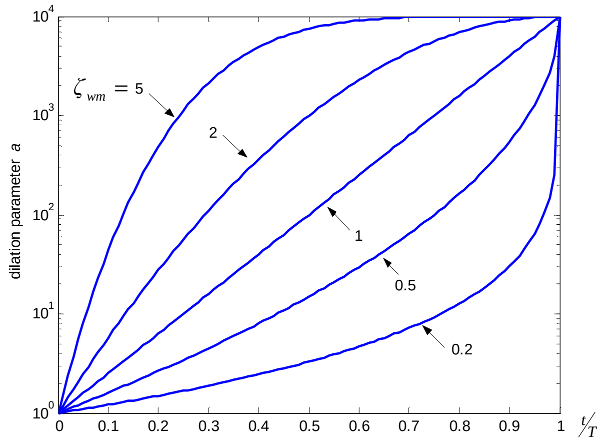

Fig. 5. Effect of the shape parameter ζ_{wm} to a with respect to t/T.

When $|\sigma|$ is small, it gives a smaller searching space for fine-tuning. Referring to *Property 1* of the wavelet, the total positive energy of the mother wavelet is equal to the total negative energy of the mother wavelet. Then, the sum of the positive σ is equal to the sum of the negative σ when the number of samples is large and φ is randomly generated, i.e.

$$\frac{1}{N}\sum_N \sigma = 0 \text{ for } N \to \infty, \tag{20}$$

where N is the number of samples. Hence, the overall positive mutation and the overall negative mutation throughout the evolution are nearly the same in a statistical sense. This property gives better solution stability (smaller standard deviation of the solution values upon many trials). As over 99% of the total energy of the mother wavelet function is contained in the interval [–2.5, 2.5], φ can be generated from [–2.5, 2.5] randomly. The value of the dilation parameter a is set to vary with the value of t/T in order to meet the fine-tuning purpose, where T is the total number of iteration and t is the current number of iteration. In order to perform a local search when t is large, the value of a should increase as t/T increases so as to reduce the significance of the mutation. Hence, a monotonic increasing function governing a and t/T is proposed as follows.

$$a = e^{-\ln(g)\times\left(1-\frac{t}{T}\right)^{\zeta_{wm}} + \ln(g)} \tag{21}$$

where ζ_{wm} is the shape parameter of the monotonic increasing function, g is the upper limit of the parameter a. The effects of the various values of the shape parameter ζ_{wm} to a with respect to t/T are shown in Fig. 5. In this figure, g is set as

10000. Thus, the value of a is between 1 and 10000. Referring to (14), the maximum value of σ is 1 when the random number of φ=0 and $a=1$ (t/T = 0). Then referring to (16), the vector $\bar{\mathbf{u}}_{i,g}$ has a large degree of mutation. It ensures that a large search space for the mutated vector is given at the early stage of evolution. When the value t/T is near to 1, the value of a is so large that the maximum value of σ will become very small. For example, at t/T =0.9 and $\zeta_{wm}=1$, a = 400; if the random value of φ is zero, the value of σ will be equal to 0.0158. A smaller searching space for the mutated vector is given for fine-tuning.

After the operation of wavelet mutation and crossover, a new population is generated. This new population will repeat the same process. Such an iterative process will be terminated when a defined number of iteration is met.

III. Benchmark Test Functions and Results

A. Benchmark test functions

A suite of eight benchmark test functions [11] are used to test the performance of the proposed WMWC-DE. Many different kinds of optimization problems are covered by these functions, which can be divided into three categories. The first category covers the unimodal functions f_1, f_2 and f_3 that are symmetric with a single minimum. The second one covers the multimodal functions f_4 and f_5 with only a few local minima. The last one covers the multimodal functions f_6, f_7 and f_8 with many local minima. The details of these functions are shown in Table 1.

B. Experimental Setup

The performance of SDE [1], DE with wavelet mutation, and the proposed WMWC-DE are evaluated by finding the minimum values of the benchmark test functions. The following simulation conditions are used:

- The shape parameter of the wavelet mutation (ζ_{wm}): It is chosen by trial and error through experiments for good performance for all functions. ζ_{wm} =1 is used for all functions.
- Initial population: It is generated uniformly at random.
- Crossover probability constant: *Cr* = 0.5

The number of iteration for all algorithms and the values of *F* for SDE are given in Table 2.

C. Results and Analysis

In this section, the simulation results for the 8 benchmark test functions are given to show the merits of the WMWC-DE. All results shown are averaged data out of 50 trials.

Table 2. Number of iteration and the value of *F* for SDE.

Test function	No. of iteration	Fixed F Weight
Rosenbrock's function	25	0.85
Quartic function	60	0.5
Easom's function	200	0.5
Maxican hat function	20	0.5
Six-hump camel back function	20	0.5
Generalized Griewank's function	100	0.5
Generalized Ackley's function	500	0.5
Schwefel's function	500	0.5

Table 1. Benchmark Test Functions.

Test function	Domain range	Optimal point
Rosenbrock's function $f_1(\mathbf{x})=\sum_{i=1}^{30}\left[100\left(x_{i+1}-x_i^2\right)^2+\left(x_i-1\right)^2\right]$	$-2.048\le x_i\le 2.048$	Min(f_1)= f_1([1, ..., 1])=0
Quartic function $f_2(\mathbf{x})=\sum_{i=1}^{30}\left[ix_i^4\right]$	$-1.28\le x_i\le 2.56$	Min(f_2)= f_2([1, ..., 1])=0
Easom's function $f_3(\mathbf{x})=-\cos(x_1)\cdot\cos(x_2)\cdot\exp(-((x_1-\pi)^2+(x_2+\pi)^2))$	$-300\le x_1,x_2\le 300$	Min(f_3)= $f_3([\pi,\pi])=-1$
Maxican hat function $f_4(\mathbf{x})=-\frac{\sin(x_1)\sin(x_2)}{x_1x_2}$	$-5\le x_1,x_2\le 15$	Min(f_4)= $\lim_{x\to[0,0]}(\mathbf{x})=-1$
Six-hump camel back function $f_5(\mathbf{x})=4x_1^2-2.1x_1^4+\frac{1}{3}x_1^6+x_1x_2+4x_2^2+4x_2^4$	$-5\le x_1,x_2\le 5$	Min(f_5)= f_5([−0.08983, 0.7126]= f_5([0.08983, −0.7126]) = −1.0316
Generalized Griewank's function $f_6(x)=\frac{1}{4000}\sum_{i=1}^{30}x_i^2-\prod_{i=1}^{30}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	$-1200\le x_i\le 600$	Min(f_6)= $f_6(\mathbf{0})=0$
Generalized Ackley's function $f_7(\mathbf{x})=-20\exp\left(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{30}x_i^2}\right)-\exp\left(\frac{1}{30}\sum_{i=1}^{30}\cos 2\pi x_i\right)+20+e$	$-64\le x_i\le 32$	Min(f_7)= $f_7(\mathbf{0})=0$
Schwefel's function $f_8(x)=\sum_{i=1}^{30}(x_i\sin(\sqrt{\lvert x_i\rvert}))$	$-500\le x_i\le 500$	Min(f_8)= f_8([420.9687, ..., 420.9687])= −12569.5

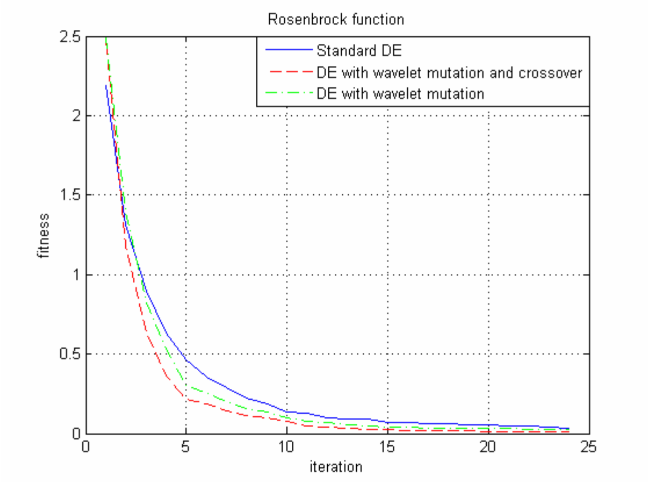

Fig. 6. The fitness of the Rosenbrock function.

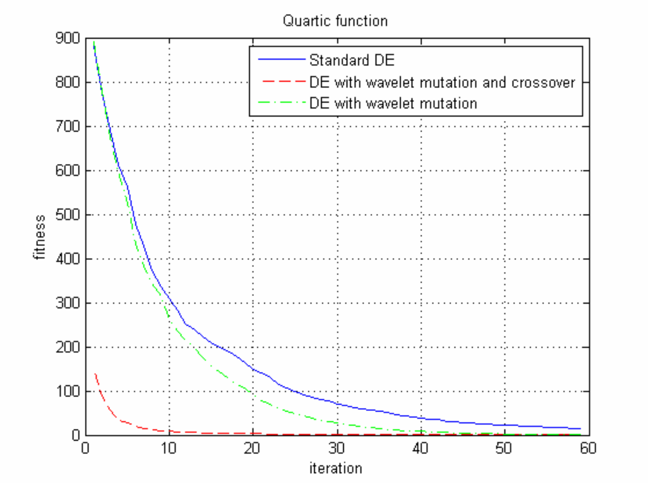

Fig. 7. The fitness of the Quartic function.

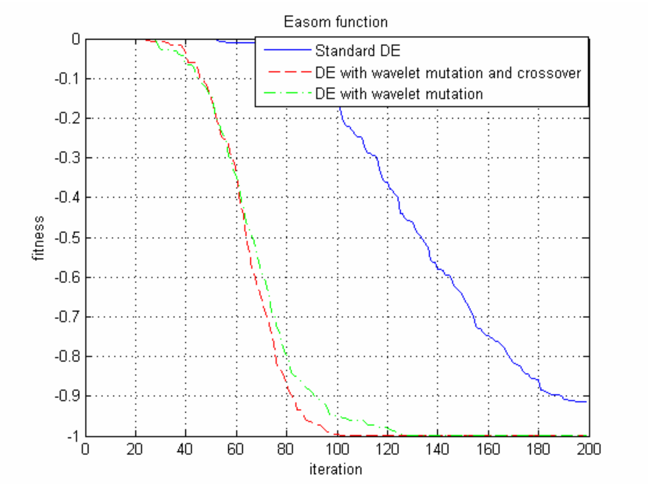

Fig. 8. The fitness of the Easom's function.

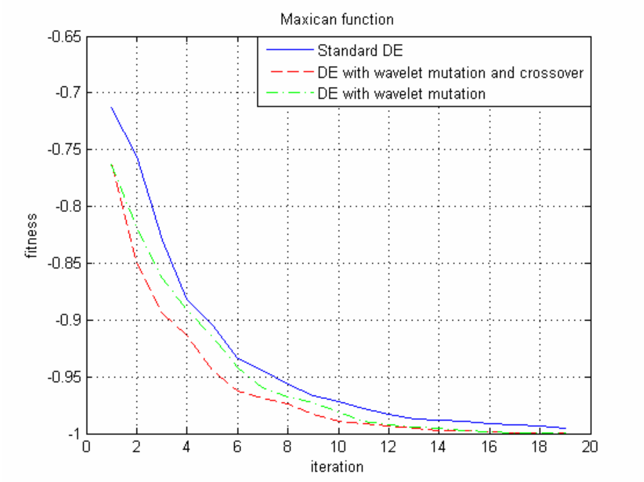

Fig. 9. The fitness of the Maxican hat function.

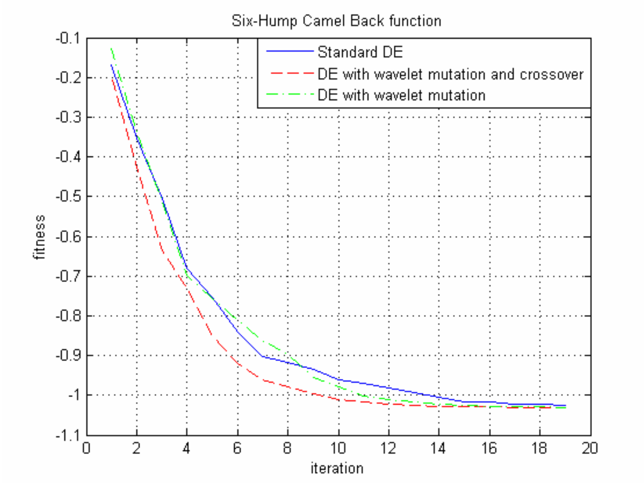

Fig. 10. The fitness of the Six-hump camel back function.

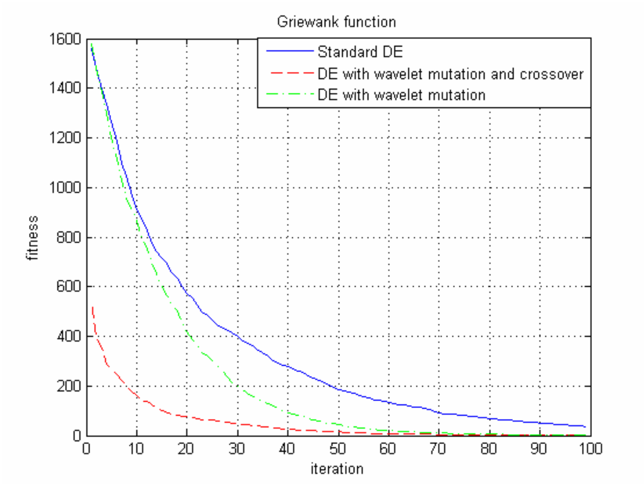

Fig. 11. The fitness of the Generalized Griewank function

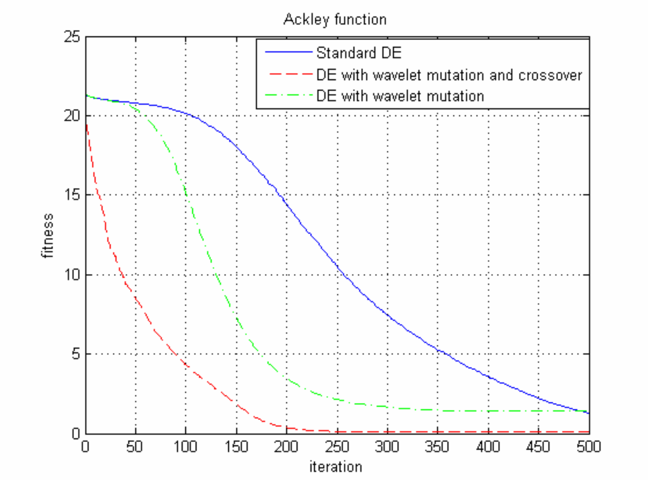

Fig. 12. The fitness of the Ackley function.

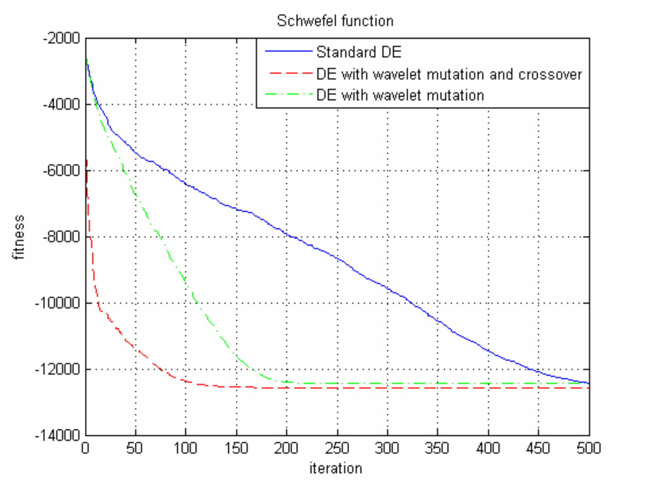

Fig. 13. The fitness of the Schwefel's function.

1. Unimodal functions

Function f_1 is the Rosenbrock function, which is also called the Banana function. The global minimum of these functions is inside a long, narrow, parabolic shaped flat valley. Owing to the smooth and symmetric characteristic of f_1, the main purpose of testing is to measure the convergence rate of the searching algorithms. It is probably the most widely used test function. The result is shown in Fig. 6. The convergence rate of the proposed WMWC-DE is a bit higher than that of SDE. When using the proposed WMWC-DE, the solution quality is increased when the number of iteration increases. As there is only one minimum within the solution space, nearly all the population will move towards that minimum.

Function f_2 is the Quartic function. Since it is a polynomial of even degree, it approaches the same limit when the argument goes to positive or negative infinity. Thus the function has a global minimum. The result is shown in Fig. 7. We can see that the convergence rate of the proposed WMWC-DE is much greater than that of SDE. After around 20 times of iteration, the proposed method is able to reach the minimum.

Function f_3 is the Easom function where the global minimum is near a small area relative to the search space. The function was inverted for minimization. The result is shown in Fig. 8. For this function, the convergence rate of the proposed WMWC-DE is much higher than that of SDE. While the performance of WMMC-DE is nearly the same as DE with wavelet mutation only, this experiment shows that the wavelet crossover does not take any advantage on reaching the minimum. But the wavelet crossover does not worsen the performance of DE. The solution quality is nearly the same for all algorithms when the number of iteration increases. Taking advantage of the properties of the wavelet function to control the scaling factor in the wavelet mutation, the population can be kept in the small area near the minimum point.

For unimodal functions, the proposed WDE can offer a higher rate of convergence as compared with SDE. By adopting the Morlet wavelet on controlling the scaling factor *F*, the degree of freedom of the trial vector can be increased. More vector directions would be generated during the mutation operation. Moreover, based on the fine-tuning ability of the wavelet crossover operation, the population can easily get into the small region around the global minimum.

2. Multimodal functions with a few local minima

Two multimodal functions with a few local minima are evaluated with the three algorithms. Function f_4 is the Maxican hat function and function f_5 is the six-hump camel back function. All of them contain some local minima within the searching space. The results are shown in Fig. 9 and Fig. 10. For function f_4 and f_5, it is found that all the searching methods perform similarly in reaching the optimal point. While the functions contain a few local minima, all the searching methods do not get trapped in some local minima easily. The advantage brought by the wavelet mutation and wavelet crossover to the searching is not obvious for these functions. Although the wavelet mutation and wavelet crossover do not bring significant improvement to reach the minimum of these functions, the convergence rate of the proposed WMWC-DE is still a bit higher than that of the SDE.

3. Multimodal functions with many local minima

Functions f_6 is the Generalized Griewank's function which is a multimodal function with many local minima. Griewank's function is a widely employed test function for global optimization. This function has an exponentially increasing number of local minima as its dimension increases and the locations of the minima are regularly distributed. In the experiment, the dimension of the Generalized Griewank's function is 30. In consequence, the testing function contains plenty of local minima. The tested result is shown in Fig. 11. It can be seen from this figure that if the wavelet mutation is used, the rate of convergence is much higher than that of the SDE. It shows that by adding the wavelet mutation and wavelet crossover to the DE, we can reduce the chance that the searching process is trapped in some local minima. Moreover, by introducing the wavelet crossover to DE, the searching process of WMWC-DE is capable of moving closely to the global minimum in the early iteration stage, as compared with the other two algorithms. Thanks to the property of the wavelet crossover, the effort on searching and evaluating those local minima that are far away from the global minimum is reduced.

Functions f_7 is the Generalized Ackley's function which is a continuous, multimodal function obtained by modulating an exponential function with a cosine wave of moderate amplitude. Its topology is characterized by an almost flat outer region and a central hole or peak where the modulations of the cosine wave become more and more influential. The result is shown in Fig. 12. The experiment shows that if the wavelet mutation is used with DE, the fitness of the function dropped rapidly. After 200 times of iteration, the fitness value is already close to the global minimum. But both the SDE and DE with wavelet mutation cannot reach the global minimum. By applying both the wavelet mutation and wavelet crossover, the DE can reach the global minimum point at about 250 times of iteration. It shows that WMWC-DE provides a better solution quality. Furthermore, it can be seen from Fig. 12 that if the wavelet mutation and wavelet crossover are used, the convergence rate is much higher than that of the SDE. The WMWC-DE is capable of moving closely to the global minimum at the early iteration stage. This shows the advantage of incorporating wavelet mutation and wavelet crossover on reducing the effort on searching and evaluating those local minima that are far away from the global minimum.

Functions f_8 is the Schwefel's function which is deceptive in that the global minimum is geometrically distant, over the parameter space, from the next best local minima. Therefore, the search algorithms are potentially prone to convergence in the wrong direction. The result is shown in Fig. 13. Similar to functions f_6 and f_7, if the wavelet mutation and wavelet crossover are used, the convergence rate is much higher than that of the SDE. Moreover, the WMWC-DE can move closely to the global minimum at the early iteration stage.

For multimodal functions with many local minima, the proposed WMWC-DE can significantly improve the convergence rate and the chance of reaching the global optimum as compared to SDE.

IV. Conclusion

In this paper, we proposed a new hybrid differential evolution with wavelet theory based mutation and crossover operation. In the mutation operation, we proposed an adaptive scheme on tuning the scaling factor *F* of the DE algorithm by applying the wavelet theory. In the crossover operation of DE, we proposed an adaptive scheme on modifying the trial population vectors by applying the wavelet theory. The resulting WMWC-DE takes advantage of the beneficial properties of the wavelet function to improve the solution quality and stability. The proposed method can explore the solution space more effectively in reaching the global solution. Simulation results have shown that the proposed wavelet mutation and wavelet crossover based DE is a useful algorithm to solve a suite of benchmark test functions, and offers better results in terms of convergence rate, solution quality and stability than SDE. Thanks to the properties of the wavelet, the performance and robustness of DE are improved.

Acknowledgement

The work described in this paper was substantially supported by a grant from The Hong Kong Polytechnic University (Project No. RP9L).

References

[1] R. Storn and K. Price "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.

[2] S. Paterlini and T. Krink, "High performance clustering with differential evolution," in *Proc. IEEE Congress on Evolutionary Computation*, vol. 2, 2004, pp. 2004–2011.

[3] J.H. van Sickel, K.Y. Lee, and J.S. Heo, "Differential evolution and its applications to power plant control," in *Proc. Intelligent Systems Applications to Power Systems 2007*, (ISAP 2007), 5-8 Nov. 2007, pp.1 – 6.

[4] B. Babu and R.Angira, "Optimization of non-linear functions using evolutionary computation," in *Proc. 12th ISME International Conference on Mechanical Engineering*, India, 2001, pp. 153–157.

[5] L. Fogel, "Evolutionary programming in perspective: The top-down view," *Computational Intelligence: Imitating Life.* Piscataway, NJ: IEEE Press, 1994.

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

[7] I. Daubechies, *Ten lectures on Wavelets*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1992.

[8] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th International Symposium on Micro Machine and Human Science*, Nagoya, Oct. 1995, pp.39-43.

[9] S.H. Ling, C.W. Yeung, K.Y. Chan, H.H.C. Iu, and F.H.F. Leung, "A new hybrid particle swarm optimization with wavelet theory based mutation operation," in *Proc. 2007 IEEE Congress on Evolutionary Computation (CEC 2007)*, Singapore, Sep. 25-28, 2007, pp. 1977-1984.

[10] C.W. Yeung, S.H. Ling, Y.H. Chan, S.H. Ling, and F.H.F Leung "Restoration of half-toned color-quantizes images using particle swarm optimization with wavelet mutation" in *Proc. IEEE TENCON*, Nov. 2008, pp. 19 – 21.

[11] X. Yao and Y. Liu, "Evolutionary programming made faster," *IEEE Trans. Evolutionary Computation*, vol. 3, no. 2, pp. 82-102, July 1999.

[12] U.K. Chakraborty (Ed.), *Advances in Differential Evolution*. Springer, Heidelberg, 2008.