2009 International Conference on Machine Learning and Computing
IPCSIT vol.3 (2011) © (2011) IACSIT Press, Singapore

Modified the Performance of Differential Evolution Algorithm with Dual Evolution Strategy

Ying-Chih Wu [1], Wei-Ping Lee [1], Ching-Wei Chien [1]

[1] Chung Yuan Christian University, Information Management Department, Chun li, Taiwan, 32023.

Abstract. Differential Evolution (DE) is one of the novel algorithms of evolution computation. Although it performs superiorly, DE has several disadvantages. In this study, we proposed the construction of a novel DEPSO algorithm in DE and Particle Swarm Optimization (PSO). DEPSO is a strategy of Dual Evolution (DES) based on the master-apprentice mechanism for sharing information. During the iteration, between the two algorithms can be iterative operation to improve the drawbacks "easy to drop into region optimum" moreover increasing the performance to obtain the advantage of accuracy solving and stable convergence.

Keywords: Differential Evolution (DE), Particle Swarm Optimization (PSO), Differential Evolution Particle Swarm Optimization (DEPSO), Dual Evolution Strategy (EDS)

1. Introduction

Evolutionary Computation (EC) uses ideas inspired from the behavior of community-based animals such as ants, bees and birds. Furthermore researchers proposed algorithms including DE and PSO through observing their cooperation mechanism and intelligence behavior. Evolution computation is extensively used in traditional algorithms that cannot be solved and apply to data mining, network optimization, scheduling, route planning and decision support.

PSO is inspired from the foraging behavior of birds [1], because of fast convergence, fewer parameters setting, and the easiness to implement. Therefore, PSO has been popular in different fields. Even though PSO is efficient, it also has some critical problems such as premature convergence and easily drops into regional optimum.

DE has several advantages: it can search randomly, requires only fewer parameters setting, high performance and applicable to high-dimensional complex optimization problems. But similar to PSO, DE has several drawbacks including unstable convergence in the last period and easy to drop into regional optimum.

Compared with else evolutionary computation, DE and PSO have advantage respectively. In this study, based on DE and PSO to construct in a dual evolution algorithm DEPSO, rely on information sharing and storing to reserve the advantages of two algorithms moreover complementary between DE and PSO for enhancing total performance.

2. Literature Review

2.1. Particle Swarm Optimization (PSO)

Particle Swarm Optimization, one of the evolutionary computation tools, was first published in 1995 by

+ Corresponding author. Tel.: +886919145365; fax: +88632655499.
E-mail address: colles.peter@msa.hinet.net.

Kennedy and Eberhart [1]. The principle concept is based on the swarm intelligence (SI) of birds or fishes. When particles move in an N-dimension space, every particle position represents a feasible solution. During the generation, particle will obtain the best solution by their own experience called regional optimal solution (pbest), according to compare with all particle's pbest to find the appropriate solution become global optimal solution (gbest), simultaneously all of the particle will be converge directly to the gbest. Movement vector will be influenced by the following factors: (1) inertia vector (2)pbest (3)gbest.

The basic formula of PSO is given by:

$$V_{id}(t+1)=w*V_{id}(t)+C_1*rand()*(P_{id}-X_{id})+C_2*rand()*(P_{gd}-X_{id}) \quad (1)$$

$$X_{id}(t+1)=X_{id}(t)+V_{id}(t+1) \quad (2)$$

Here, ω is the inertia weight [2], $V_{id}(t)$ represents the velocity of particle, $V_{id}(t+1)$ represents the new velocity of particle, X_{id} represent the new position of particle, C_1 and C_2 are "cognitive" and a "social" components, and $rand()$ is a random number between 0 and 1.

The procedure of PSO is discussed in the following:

Step 1: Initialization, including parameter setting, initialization of the velocity and position of particle.

Step 2: Evaluate the fitness.

Step 3: Evaluate the new position's fitness; for each particle, if the fitness of new particle is better than the original particle, swap it.

Step 4: Compare with all pbest to find gbest.

Step 5: Renew velocity and position by equations (1) and (2).

Step 6: If the termination condition is failed to reach, go back to step 2, or output the optimal solution.

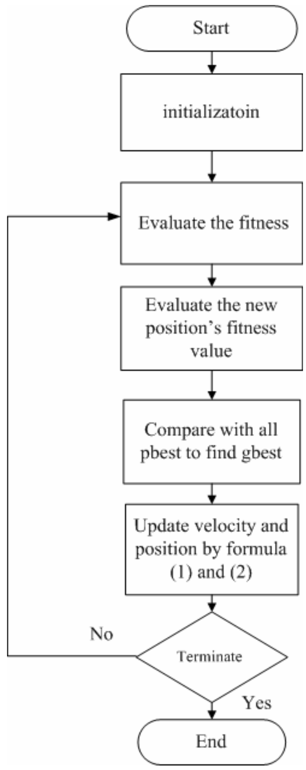

Fig. 1: The procedure of Particle Swarm Optimization.

2.2. Differential Evolution (DE)

Differential Evolution is announced in 1995 by Price and Storn, and its superior performance in solving complex problems was revealed in the 1996 IEEE international evolutionary computation conference[3,13-14].DE is based on individual's difference, utilize random research in solution space, further utilize the mechanism "mutation", "recombination", "selection" to compute every individuals to obtain appropriate individual. DE used the information between the difference in individuals to lead to search, thus the result of search is more unstable [8]. With a comparison of DE and PSO is given in the following:

Table. 1: The advantage and weakness of DE.

	DE	PSO
Advantage	● Keep the multiplicity of population ● Enhance the capacity of local search	● The convergence is quick ● Have the character of memory
Weakness	● The convergence is unstable ● Easy to drop into the pbest	● The multiplicity of population is not enough ● The convergence is untimely ● Easy to drop into the pbest

DE is similar to Genetic Algorithm, and the main procedure is discussed in the following [4]:

- Initialization：Setup the parameters and initialize the Target Vector.

$$x_{j,i,0} = rand_j(0,1) * (b_{j,U} - b_{j,L}) + b_{j,L} \quad (3)$$

- Mutation ：The common strategies of DE are via the formula of mutation vector to produce change. They are such as the following table2. The strategy symbol is DE/x/y. x means the type of variance vector and y means the number of variance vector.

Table. 2: The advantage and weakness of DE

Symbol	Formula	Description
DE/rand/1	$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G})$ (4)	For extending the search space, random selected three vectors $X_{r1,G}$, $X_{r2,G}$, *and* $X_{r3,G}$ using equation (4) to obtain Donor Vector as $V_{i,G+1}$
DE/best/1	$V_{i,G+1} = X_{best,G} + F(X_{r2,G} - X_{r3,G})$ (5)	Based on the DE/rand/1, this strategy use the present best particle $X_{best,G}$ to replace $X_{r1,G.}$

- Recombination: The donor vector will change the information with the target vector randomly. A new vector "trial vector" called $u_{i,G+1}$ will be generated after recombination. Use the following formulation to decide in the iteration $\boldsymbol{j}$ the component $\boldsymbol{i}$ compose from target vector $\boldsymbol{x_i}$ or donor vector $\boldsymbol{v_i}$.

$$u_{j,i,G+1} = \begin{cases} v_{j,i,G+1} & \text{if rand} \le CR \\ x_{j,i,G} & \text{if rand} > CR \end{cases} \tag{6}$$

- Selection: After the above mechanism, compare with Trial Vector and Target Vector to select which vector can be reserved into the next generation.

$$X_{i,G+1} = \begin{cases} u_{i,G+1} & \text{if } F(u_{i,G+1}) \leqq F(X_{i,G}) \\ X_{i,G} & \text{otherwise} \end{cases} \tag{7}$$

The following steps are involved:

Step 1: Initialization.

Step 2: Apply formula (3) to select initial Target Vector $X_{i,G}$ randomly.

Step 3: Evaluate fitness.

Step 4: Mutation: Selected three vectors according to formula (4) or (5) to mutate to generate Donor Vector.

Step 5: Recombination: Change information between the Target vector and the Donor Vector by formula (6) and obtain the Trial Vector.

Step 6: Compare with Target Vector and Trial Vector selected by fitness to determine which can be reserve into next generation.

Step 7: If the termination condition is failed to reach, go back to step 2, or output the optimal solution.

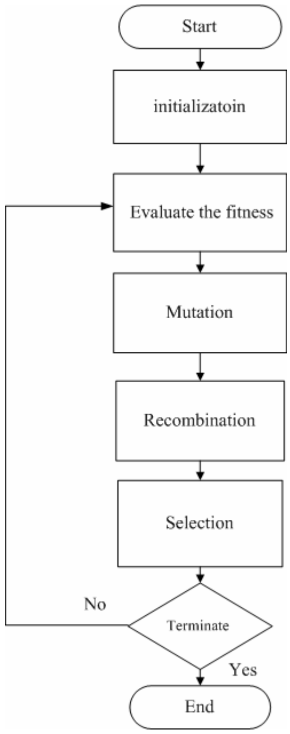

Fig. 2: The procedure of Differential Evolution.

2.3. Dual Evolution Strategy (DES)

Xing Xu et al. first reported the dual evolution structure in 2003, DE-based, PSO-supplemented called DE-SI. DE-SI is similar with traditional DE, utilize the concept "combine algorithms" and improve PSO'S velocity renew formula for obtain fast convergence. Omran et al. remained PSO-based, DE-supplemented to involve DE's mechanism "Mutation", "Recombination", "Selection" into PSO to improve the defect "premature convergence" of PSO.

Previous Studies most combine PSO and DE to enhance the performance and apply to Practical application. According to different structure, divided into

1. PSO-based, DE-supplemented
2. DE-based, PSO-supplemented

In this study, we proposed a dual evolution by iteration alternating strategy, brush aside previous studies used combination to merge DE and PSO [5-12]. Though certain iteration to share particle's information, sharing better result to another by iteration alternating.

On the one hand, acquire the superiority of DE and PSO simultaneously, on the other hand, complement each other's deficiencies.

Authors submitting a manuscript do so on the understanding

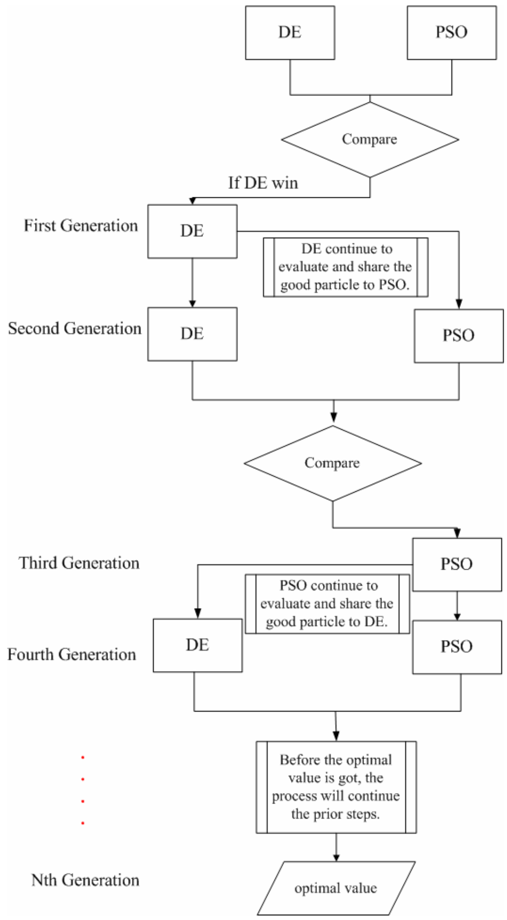

Fig. 3: The procedure of DEPSO

that if the manuscript is accepted for publication, copyright for the article, including the right to reproduce the article in all forms and media, shall be assigned exclusively to the Publisher.

3. Research Methods and Experiment Design

3.1. Experiment Design

In this study, we proposed DEPSO algorithm is constructed on traditional DE/rand/1 and PSO, utilizing DES to improve the original structure of DE.

The procedure of DEPSO is following:

Step 1：initialization.

Step 2：DE and PSO operation separately, and compare with the fitness, selected better one start to evolution.

Step 3：In odd generation, above algorithm sharing the good particle has been sorting to another algorithm to increasing the performance and convergence by better result.

Step 4：In even generation, DE and PSO will be competing and select better algorithm into sharing mechanism.

Step 5：Failed to reach the terminate condition back to step 2, or output optimal solution.

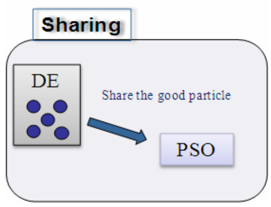

Fig. 4: sharing strategy

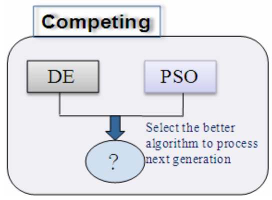

Fig. 5 competing strategy

Illustrate with Step 3 and 4 separately:

- Sharing: In odd generation, share information to another algorithm as master- apprentice. In this study, the better result from present algorithm to play the role of particle sharing, approach to learning to enhance the present result.
- Competing: In even generation, the master will compete with apprentice. Then compare with the result of DE and PSO to determine which can be the master in the next generation.

3.2. Parameter setup and Benchmark Function

In this study, unless emphasize particular parameter setup. Otherwise accepted the following parameter setup and we focus on improving the performance of DE and rely on to evaluate the performance by four benchmark functions. Four benchmark functions of its theoretical minimum value are 0. The benchmark function and parameter setup are described in the following:

Table. 3: The parameter setup.

Name	Value	Name	Value
Number of vector (NP)	50	C_1	1.4
Elite rate (elite)	0.2	C_2	1.4
Dimension (dim)	30 , 60	Search velocity (search v)	10
Mutation vector (F)	0.5	inertia weight (w)	0.9-0.4
Crossover rate (CR)	0.9	Repeat times (test_times)	200,100
Upper bound (now_up)	100	Maximum of iteration (iter)	1000
Lower bound (now_down)	-100		

Table. 4: The benchmark function.

Name	Multi-modality	Single modality	Formulation
Sphere function $f1$	No	Yes	$f_1(x)=\sum_{i=1}^{n}x^2$
Rosenbrock function $f2$	No	Yes	$f_2(x)=\sum_{i=1}^{n-1}(100(x_{i+1}-x_i^2)^2+(x_i-1)^2)$
Rastrigin function $f3$	Yes	No	$f_3=\sum_{i=1}^{n}(x_i^2-10\cos(2\pi x_i)+10)$
Griewank function $f4$	Yes	No	$f_4=\frac{1}{4000}\sum_{i=1}^{n}(x_i-100)^2-\prod_{i=1}^{n}\cos(\frac{x_i-100}{\sqrt{i}})+1$

4. Experimental Results

In the experiment of this study will be compare with DE/rand/1, DE/best/1, PSO and DEPSO to confirm with their own stability and the best experimental result, and ultimately through the experimental result analysis their performance in high dimensions(30 dimension, 60 dimension) complex function.

In this study, we improve differential evolution by dual evolution structure. Dual evolution strategy is first time implement into DE. Compare with DEPSO, DE/rand/1, DE/best/1 and PSO in solving efficiency of benchmark functions. According to the experiment result, DEPSO represents considerable extent increasing in single and multiple modality functions moreover enhance the convergence to acquire stable convergence by importing dual evolution strategy into differential evolution.

4.1. Benchmark function: Sphere (f1)

According to the experimental result of Sphere function, DEPSO has significant reinforcement than else algorithms, but in 60-dimension was unstable than 30-dimension. For above result, DEPSO confirm the convergence progress is stable and significant.

4.2. Benchmark function: Rosenbrock (f2)

Because Rosenbrock function is a complex function of single-modality, therefore these algorithms cannot obtain superiority performance, but DEPSO in 30 and 60 dimension still represents stable convergence and better result.

4.3. Benchmark function: Rastrigrin (f3)

In Rastrigrin function, DEPSO provides better result in 30 dimension, but has slight difference with DE/best/1 in 60 dimension, but DEPSO still represent more stable performance than else algorithms**.**

4.4. Benchmark function: Griewank (f4)

DEPSO is powerful performance and stable convergence, regardless 30 or 60 dimension DEPSO is superiority in function Griewank.

TABLE 5. Experiment Result of Sphere (*f1*)

f1	Dimension	DE/rand/1	DE/best/1
Avg	30	4.879650e-010	4.236681e-016
	60	2.218372e-002	1.932865e-019
Best	30	2.045238e-010	5.664503e-017
	60	1.259653e-004	5.920183e-020
	Dimension	PSO	DEPSO
Avg	30	5.376984e-007	2.130045e-026
	60	1.843294e-011	8.930297e-027
Best	30	1.283394e-011	1.216424e-035
	60	7.135705e-012	8.666767e-65

TABLE 6. Experiment Result of Rosenbrock (*f2*)

f2	Dimension	DE/rand/1	DE/best/1
Avg	30	1.290789e+002	6.460700e+001
	60	1.195137e+002	1.286586e+002
Best	30	4.479539e+001	2.831109e+000
	60	6.195290e+001	4.604165e+001
	Dimension	PSO	DEPSO
Avg	30	6.765685e+001	2.633769e+001
	60	5.703899e+002	5.714797e+001
Best	30	8.236947e+000	2.522003e+001
	60	2.129353e+002	5.521501e+01

TABLE 7. Experiment Result of Rastrigrin (*f3*)

f3	Dimension	DE/rand/1	DE/best/1
Avg	30	4.499533e+000	5.372812e-001
	60	1.035306e+002	3.872560e+001
Best	30	4.960861e-001	2.644430e-008
	60	8.537815e+001	2.665797e+000
	Dimension	PSO	DEPSO
Avg	30	6.539890e+001	1.744925e-001
	60	2.413862e+002	9.095539e+001
Best	30	2.805312e+001	0
	60	1.337377e+002	6.997933e+01

TABLE 8. Experiment Result of Griewank (*f4*)

f4	Dimension	DE/rand/1	DE/best/1
Avg	30	8.943127e-010	8.499917e-004
	60	1.161345e-012	4.190023e-004
Best	30	2.645439e-011	2.705168e-002
	60	2.832179e-013	0
	Dimension	PSO	DEPSO
Avg	30	1.384572e-002	1.155985e-011
	60	1.289793e-002	9.547918e-016
Best	30	3.339440e-012	0
	60	9.445777e-013	0

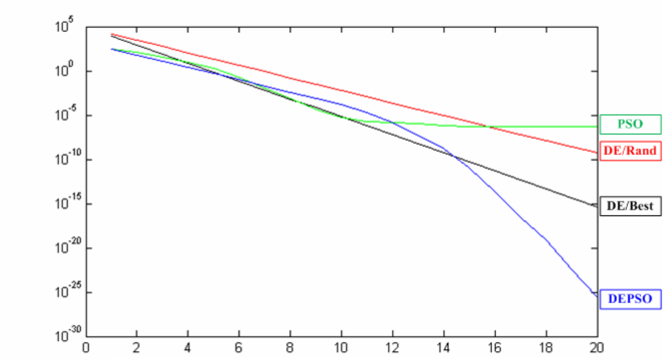

Fig. 6: *Sphere(f1)* - 30 dimension

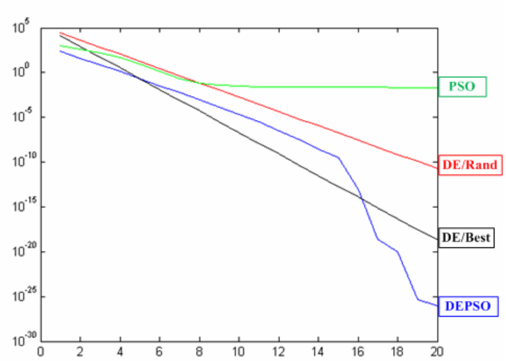

Fig. 7: *Sphere(f1)* - 60 dimension

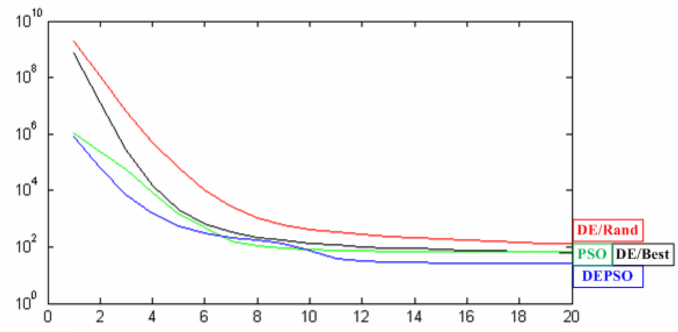

Fig. 8: *Rosenbrock (f2)* - 30 dimension

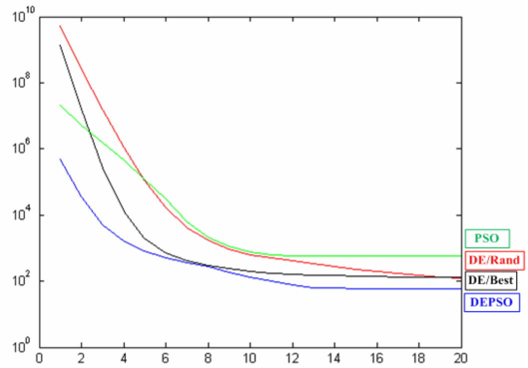

Fig. 9: *Rosenbrock (f2)* - 60 dimension

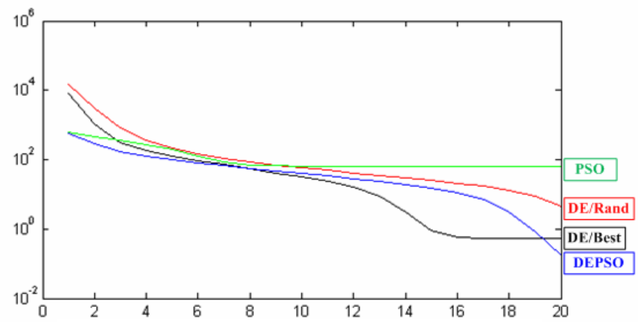

Fig. 10: *Rastrigrin (f3)* - 30 dimension

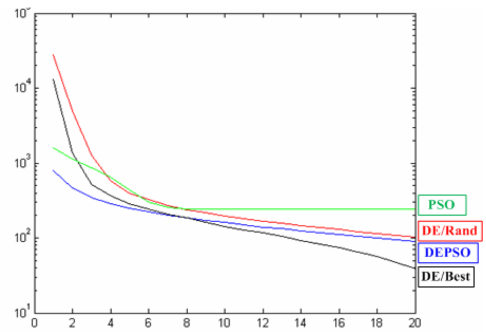

Fig. 11: *Rastrigrin (f3)* - 60 dimension

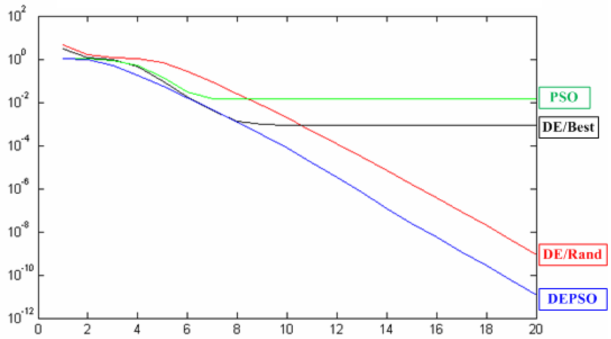

Fig. 12: *Griewank (f4)* - 30 dimension

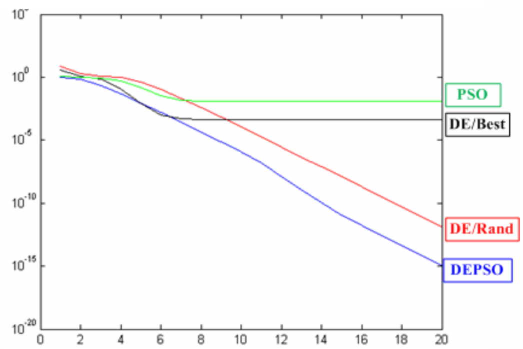

Fig. 13: *Griewank (f4)* - 60 dimension

5. Conclusion

Although DE and PSO have superior performance, there some drawbacks like dropping into regional optimum needs to be rectified. In this study, we proposed the structure of dual evolution to improve differential evolution, approach to sharing information increasing performance. During dual evolution, the performance of two algorithms against different functions needs to be compared and then determine which first carried out in accordance with. According to master-apprentice mechanism, selected better algorithm presently to share information to another algorithm, moreover confirm the accuracy of solving problems and increasing the convergence performance. Accordingly, information exchange enhances exploration and maintains diversity of the algorithm. Moreover, it decreases the probability of dropping into regional optimum. In the experiment design, DEPSO is constructed on traditional DE/rand/1 and PSO. We didn't use DE/best/1 for DEPSO's foundation, because DE/best/1 easy to drop into regional optimum and effect the performance.

According to the experiment result, DEPSO is more efficiency than DE/rand, DE/best, PSO, and proof the dual evolution structure is effectiveness; moreover approach to the result of average represents more stability by dual evolution. Follow-up experiment will investigate master-apprentice mechanism and estimate how to alternate algorithms in right occasion for enhancing the whole performance. DEPSO expected to improve the drawbacks of differential evolution such as unstable convergence and easy to drop into regional optimum, and apply to application such as data mining, scheduling and route planning. In the future, one may try to import modified DE or PSO into this dual evolution structure to enhance the performance further.

6. References

[1] J. Kennedy and R. C. Eberhart, ”Particle swarm optimization,” *Proceedings of IEEE International Conference on Neural Networks, Piscataway,* NJ. , 1995, pp. 1942-1948.

[2] Y. Shi and R. C. Eberhart, ”A modified particle swarm optimizer,” *Proceedings of IEEE International Confernce on Evolutionary Compution, Piscataway, NJ.* , 1998, pp. 69-73.

[3] R. Storn and K. Price, “Differential evolution–A simple efficient adaptive scheme for global optimization over continuous spaces,” Technical report, California: International Computer Science Institute, *Berkeley*, 1995.

[4] D. G. Mayer, and A. A. Archer, “Differential evolution – an easy and efficient evolutionary algorithm for model optimisation,” *Agricultural Systems,* vol. 83, 2005, pp. 315–328.

[5] M. G. H. Omran, A. P. Engelbrecht, and A. Salman, “Differential Evolution Based Particle Swarm Optimization,” *Proc. the 2007 IEEE Swarm Intelligence Symposium*, Apr. 2007, pp. 112–119.

[6] W. J. Zhang and X. F. Xie, “DEPSO: Hybrid Particle Swarm with Differential Evolution Operator,” *Proc. IEEE International Conference on Systems, Man and Cybernetics,* 2003, pp. 3816–3821.

[7] X. Xu, Y. Li, S. Fang, Y. Wu, and F. Wang, “A novel differential evolution scheme combined with particle swarm intelligence,” *2008 IEEE Congress on Evolutionary Computation, Jun. 2008*, pp. 1057-1062.

[8] Z.-F. Hao, G.-H. Guo, and H. Huang, “A Particle Swarm Optimization Algorithm with Differential Evolution,” *2007 IEEE International Conference on Machine Learning and Cybernetics*, vol. 2, Aug. 2007, pp. 1031-1035.

[9] D. Ning, W. Zhang, and B. Li, “Differential evolution based particle swarm optimizer for neural network learning,” *2008. 7th World Congress on Intelligent Control and Automation*, Jun. 2008, pp. 4444-4447.

[10] L. Bipul and K. V. Ganesh, “Differential evolution particle swarm optimization for digital filter design,” *2008 IEEE Congress on Evolutionary Computation*, Jun. 2008, pp. 3954-3961.

[11] R. Xu, G. K. Venayagamoorthy, and D. C. Wunsch, “Modeling of gene regulatory networks with hybrid differential evolution and particle swarm optimization,” *ESCI on Neural Networks* vol. 20, Oct. 2007, pp. 917-927.

[12] L.-J. Luan, L.-J. Tan, and B. Niu, “A Novel Hybrid Global Optimization Algorithm Based on Particle Swarm Optimization and Differential Evolution,” *Information and Control*, vol. 36, no. 6, Dec. 2007, pp. 708-714.

[13] R. Storn and K. Price, “Minimizing the real functions of the ICEC'96 contest by Differential Evolution,” *Int. Conf. on Evolutionary Computation, Nagoya, Japan*, 1996, pp. 842-844.

[14] K. Price, “Differential Evolution: A Fast and Simple Numerical Optimizer,” *1996 Biennial Conference of the North American Fuzzy Information Processing Society,* Jun. 1996, pp.524-527.