Differential Evolution Algorithm with Interval Type-2 Fuzzy Logic for the Optimization of the Mutation Parameter

Patricia Ochoa, Oscar Castillo and José Soria

Abstract In this paper we propose using interval type-2 fuzzy logic for the optimization of parameters the form dynamic using the Differential Evolution algorithm. For this particular work we use Benchmark mathematical functions for the experiments that were performed adhering to the rules of the competition for the IEEE Congress on Evolutionary Computation (CEC) benchmark set of 2015. We are presenting a comparison against the winning paper of the competition IEEE Congress on Evolutionary Computation (CEC) to verify how good the proposed method Fuzzy Differential Evolution algorithm really is.

Keywords Differential Evolution algorithm · Fuzzy Differential Evolution Type-2 fuzzy logic

1 Introduction

The use of algorithms based on nature has become very common in evolutionary computation and metaheuristics. In this paper we propose to use one of these algorithms, in particular the Differential Evolution integrating fuzzy logic for dynamically adapting its parameters.

Differential Evolution (DE) is one of the latest evolutionary algorithms that has been proposed in the literature. It was created in 1994 by Price and Storn in an attempt to solve the Chebychev polynomial problem. The following years after that these two authors also proposed the DE for optimization of nonlinear and non-differentiable functions on continuous spaces [20].

Fuzzy logic or multi-valued logic is based on the fuzzy set theory proposed by Zadeh in 1965, which can help us with modeling expert knowledge, through the use of if-then fuzzy rules. Fuzzy set theory provides a systematic calculus to deal with

P. Ochoa · O. Castillo (✉) · J. Soria
Tijuana Institute of Technology, Tijuana, BC, Mexico
e-mail: ocastillo@tectijuana.mx

linguistic information, and improves the numerical computation by using linguistic labels stipulated by membership functions [16].

In addition, in reviewing the recent literature there are papers on Differential Evolution applications that use this algorithm to solve real world problems [23, 24]. In the last years the concept of fuzzy logic has been used to adapt certain parameters in metaheuristic algorithms [15, 17, 18, 26–28], which demonstrate the importance and the improvement of these algorithms.

Regarding related works that use fuzzy logic to optimize the performance of metaheuristic algorithms we can find some relevant ones in [4, 5, 11–13].

In addition, related work with regards to the Special Session & Competition on Real-Parameter Single Objective Optimization at CEC-2015 can also be mentioned, like the works that obtained the first place: A Self-Optimization Approach for L-SHADE Incorporated with Eigenvector-Based Crossover and Successful-Parent-Selecting Framework on CEC 2015 Benchmark Set [8] (rank 1).Finally, the other works presented in the Special Session & Competition on Real-Parameter Single Objective Optimization at CEC-2015 [1–3, 6, 8, 19, 21, 22, 29, 30].

The rest of the paper is organized in the following form: Sect. 2 describes the Differential Evolution algorithm. Section 3 describes the proposed methods using the fuzzy logic approach. Section 4 presents the experimentation with the Benchmark functions. Finally, Sect. 5 offers the Conclusions.

2 The Differential Evolution Algorithm

Differential Evolution algorithm (DE) is an optimization method belonging to the category of evolutionary computation that can be applied in solving complex optimization problems. The differential evolution consists mainly of 4 steps [20]:

- Initialization
- Mutation
- Crossing
- Selection

This is the mathematical form of the Differential Evolution algorithm [13, 14]:
Population structure

$$P_{x,g} = (\mathbf{x}_{i,g}), \quad i = 0, 1, \ldots, Np, \ g = 0, 1, \ldots, g_{max} \tag{1}$$

$$\mathbf{x}_{i,g} = (x_{j,i,g}), \quad j = 0, 1, \ldots, D - 1 \tag{2}$$

$$P_{v,g} = (\mathbf{v}_{i,g}), \quad i = 0, 1, \ldots, Np - 1, \ g = 0, 1, \ldots, g_{max} \tag{3}$$

$$v_{i,g} = (v_{j,I,g}), \quad j = 0, 1, \ldots, D - 1 \tag{4}$$

$$P_{v,g} = (u_{i,g}), \quad i = 0, 1, \ldots, Np - 1, \quad g = 0, 1, \ldots, g_{max} \tag{5}$$

$$u_{i,g} = (u_{j,I,g}), \quad j = 0, 1, \ldots, D - 1 \tag{6}$$

Initialization

$$x_{j,i,0} = rand_j(0, 1) \cdot (b_{j,U} - b_{j,L}) + b_{j,L} \tag{7}$$

Mutation

$$v_{i,g} = \mathbf{x}_{r0,g} + F \cdot (\mathbf{x}_{r1,g} - \mathbf{x}_{r2,g}) \tag{8}$$

Crossover

$$U_{i,g} = (u_{j,i,g}) = \left\{ \begin{array}{ll} v_{j,i,g} & if \ (rand_j(0, 1) \leq Cr \ or \ j = j_{rand}) \\ x_{j,i,g} & otherwise. \end{array} \right\} \tag{9}$$

Selection

$$X_{i,g+1} = \left\{ \begin{array}{ll} U_{i,g} & if \ f(U_{i,g}) \leq f(X_{i,g}) \\ X_{i,g} & otherwise. \end{array} \right\} \tag{10}$$

3 Proposed Method

We propose using DE enhanced with fuzzy logic to dynamically modify the F parameter (mutation) during execution of the algorithm. We have previously work with the Differential Evolution algorithm using fuzzy logic and this method was called Fuzzy Differential Evolution, and we are now extending this previous work by using new and more complex functions to verify in more detail the efficiency of the proposed algorithm [14]. The way in which we integrate a fuzzy system algorithm can be found in Fig. 1, where we have the flowchart of the Differential Evolution algorithm and we have a component with Interval-type 2 fuzzy logic, which dynamically calculates the F parameter (mutation) to then make calculations in the algorithm. Figure 2 shows the structure for the Interval-type 2 fuzzy system, which contains one input and one output.

Figure 3 shows the input variable (Generation) that has 3 membership functions and is granulated in the Low, Medium and High values that were used by each membership function are symmetric the ranges are as follows:

- Low: $\begin{bmatrix} -0.5859 & -0.08598 & 0.4141 & -0.4193 & 0.08068 & 0.5807 \end{bmatrix}$
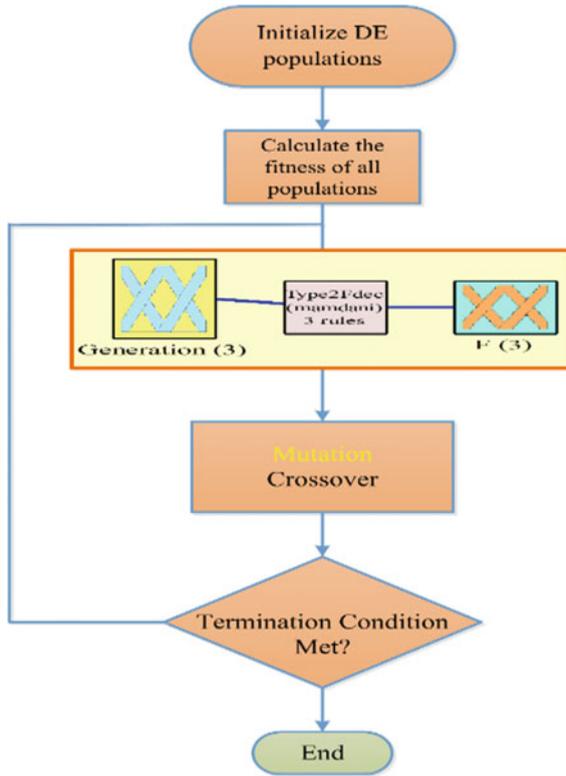
Fig. 1 Propose method with Interval-type 2 fuzzy logic
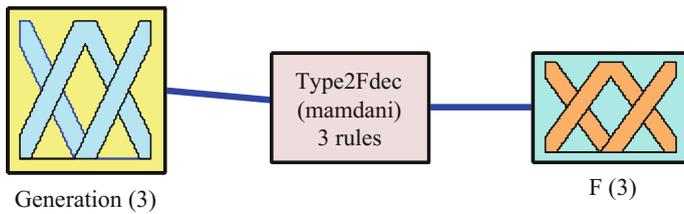
Fig. 2 Fuzzy system

- Medium: $\begin{bmatrix} -0.08333 & 0.4167 & 0.9167 & 0.08333 & 0.5833 & 1.083 \end{bmatrix}$
- High: $\begin{bmatrix} 0.4167 & 0.9167 & 1.417 & 0.5833 & 1.083 & 1.583 \end{bmatrix}$

Figure 4 presents the output variable (F) has 3 membership functions and is granulated in Low, Medium and High values that were used by each membership function are symmetric the ranges are as follows:

Fig. 3 Input generation
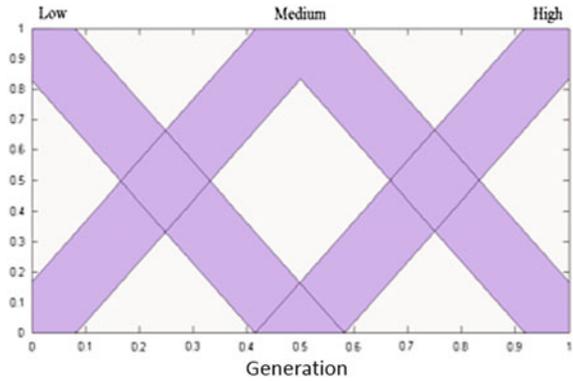
Fig. 4 Output F parameter
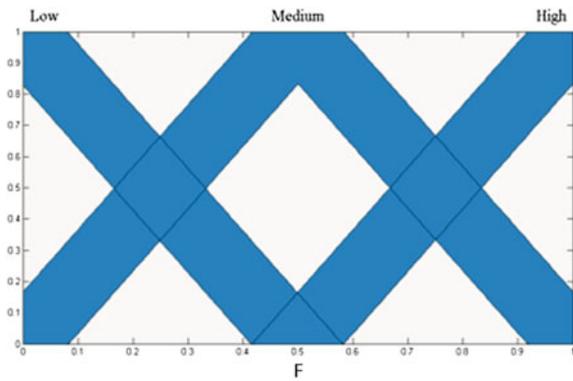
Fig. 5 Rules for the fuzzy system

1. - If (Generations is Low) then (F is High) (1)

2. - If (Generations is Medium) then (F is Medium) (1)

3. - If (Generations is High) then (F is Low) (1)

- Low: $[-0.5859 \quad -0.08598 \quad 0.4141 \quad -0.4193 \quad 0.08068 \quad 0.5807]$
- Medium: $[-0.0807 \quad 0.419 \quad 0.919 \quad 0.086 \quad 0.586 \quad 1.09]$
- High: $[0.4167 \quad 0.9167 \quad 1.417 \quad 0.5833 \quad 1.083 \quad 1.583]$

Figure 5 represents the rules of the interval-type 2 fuzzy logic and the Fig. 6 shows the surface of the interval-type 2 fuzzy system.

Fig. 6 Surface of the fuzzy system
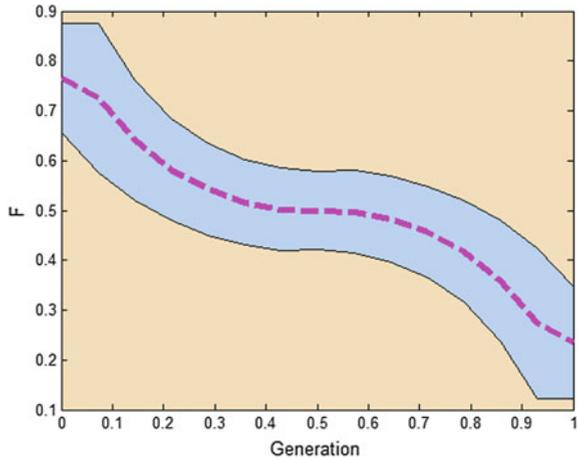
4   Results of the Experiments

In this paper we consider the set of CEC 2015 Benchmark Functions to perform the experiments and thereby be to [7, 9, 10, 25].

Below a brief description of the set of CEC 2015 Benchmark Functions is presented.

Table 1 shows the set of 15 CEC 2015 Benchmark Functions used in this paper.

For the experiments we use the metrics given by the comments of CEC'15 Benchmark Functions, are comprised of 15 minimization problems and the experiments for all functions are performed with different dimensions and for this competition are of D = 10, 30, 50, 100.

Tables 2 and 3 show the comparison between the article that obtained the first place in the competition of the CEC´15 and our proposed method [8].

Figures 7, 8, 9 and 10 represent the comparison graph for each dimension D = 10, D = 30, D = 50 and D = 100.

Table 1 Summary of the CEC'15 learning-based benchmark suite

	No.	Functions	$Fi^* = Fi$ (x^*)
Unimodal functions	1	Rotated high conditioned elliptic function	100
	2	Rotated cigar function	200
Simple multimodal functions	3	Shifted and rotated Ackley's function	300
	4	Shifted and rotated Rastrigin's function	400
	5	Shifted and rotated Schwefel's function	500
Hybrid functions	6	Hybrid function 1 ($N = 3$)	600
	7	Hybrid function 2 ($N = 4$)	700
	8	Hybrid function 3 ($N = 5$)	800
Composition functions	9	Composition function 1 ($N = 3$)	900
	10	Composition function 2 ($N = 3$)	1000
	11	Composition function 3 ($N = 5$)	1100
	12	Composition function 4 ($N = 5$)	1200
	13	Composition function 5 ($N = 5$)	1300
	14	Composition function 6 ($N = 7$)	1400
	15	Composition function 7 ($N = $ l0)	1500

Search range: $[-100,100]^D$

Table 2 Comparison between the references and the propose method with D = 10 and D = 30

	D = 10		D = 30	
	Best SPS-L-SHADE-EIG [8]	Best (FDE IT2)	Best SPS-L-SHADE-EIG [8]	Best (FDE IT2)
f1	0.00E+00	3.50E−03	0.00E+00	2.20E−02
f2	0.00E+00	1.00E+00	0.00E+00	1.11E+00
f3	0.00E+00	0.00E+00	2.00E+01	2.10E+01
f4	0.00E+00	0.00E+00	1.05E−02	5.12E+02
f5	3.12E−01	0.00E+00	6.58E+02	8.18E+03
f6	0.00E+00	0.00E+00	2.68E+01	5.18E−03
f7	0.00E+00	0.00E+00	6.23E−01	3.23E+02
f8	6.02E−08	0.00E+00	2.07E+00	2.40E−03
f9	1.00E+02	1.20E+02	1.02E+02	1.20E+02
f10	2.17E+02	2.04E+03	1.48E+02	3.25E−03
f11	2.61E–02	2.01E+02	3.00E+02	1.70E+03
f12	1.00E+02	1.34E+02	1.02E+02	2.14E+02
f13	3.03E−02	5.09E+01	2.56E−02	1.57E+02
f14	1.00E+02	1.15E+02	3.11E+04	8.35E+04
f15	1.00E+02	1.63E+01	1.00E+02	4.99E+04

Table 3 Comparison between the references and proposes method D = 50 and D = 100

	D = 50		D = 100	
	Best SPS-L-SHADE-EIG [8]	Best (FDE IT2)	Best SPS-L-SHADE-EIG [8]	Best (FDE IT2)
f1	0.00E+00	5.28E−02	0.00E+00	1.21E−01
f2	0.00E+00	2.01E+00	0.00E+00	3.81E+00
f3	2.00E+01	2.13E+01	2.00E+01	2.01E+01
f4	9.01E–05	1.06E+03	1.59E+01	2.29E+03
f5	1.38E+03	1.40E+04	3.98E+03	4.23E+04
f6	5.18E+01	3.11E−03	9.61E+02	9.82E+02
f7	6.49E+00	1.58E+03	9.06E+01	1.52E+04
f8	1.00E+01	1.35E−03	6.87E+02	7.12E+02
f9	1.03E+02	1.20E+02	1.05E+02	3.20E+03
f10	6.79E+02	2.10E−03	1.59E+03	1.78E+03
f11	3.00E+02	2.85E+03	3.01E+02	4.08E+03
f12	1.03E+02	2.97E+02	1.11E+02	3.28E+02
f13	6.99E–02	9.09E+02	5.96E–02	7.11E+02
f14	4.95E+04	2.82E+04	1.09E+05	1.15E+05
f15	1.00E+02	3.78E+05	1.00E+02	2.00E+04
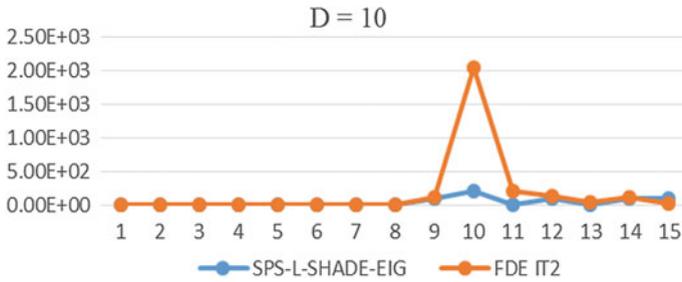
Fig. 7 Comparison with D = 10
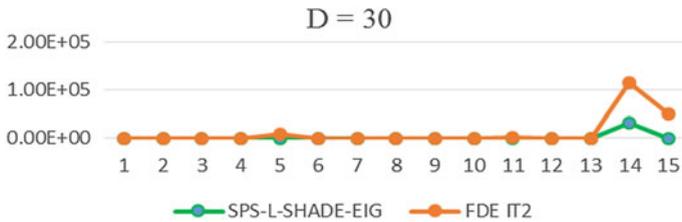
Fig. 8 Comparison with D = 30
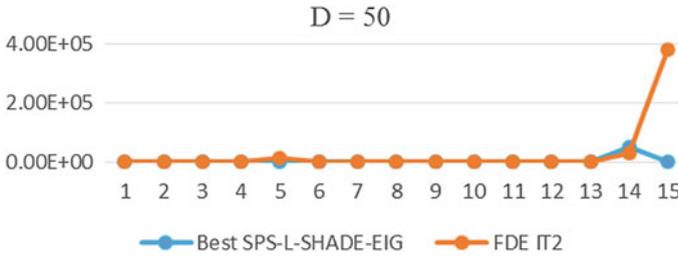
Fig. 9 Comparison with D = 50
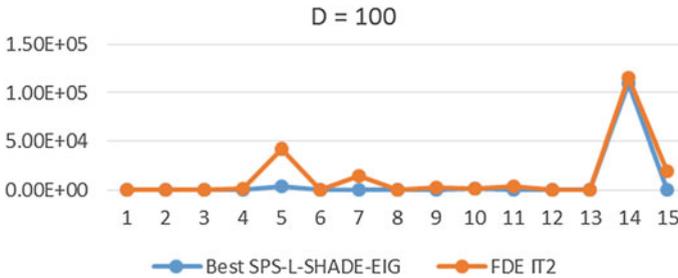
Fig. 10 Comparison with D = 100

5 Conclusions

In conclusion, we can notice that the results obtained using interval-type 2 fuzzy logic are not as good as we expected, most likely because the fuzzy system is a simple one input and one output and another part of the footprint uncertainty membership functions the input and the output is symmetric and not optimized.

We will try to improve our fuzzy system and/or we will make the footprint uncertainty trail be optimized by some algorithm and thus be able to make a comparison with statistical test to be able to affirm or reject if our proposed method is good with the functions Benchmark, But we can also conclude that our diffuse system for some functions if it is competitive and for those cases means that the uncertainty footprint is the one indicated for certain functions Benchmark CEC'15.

References

1. A. Al-Dujaili, K. Subramanian, S. Suresh, HumanCog: a cognitive architecture for solving optimization problems, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 3220–3227

2. N. Awad, M.Z. Ali, R.G. Reynolds, A differential evolution algorithm with success-based parameter adaptation for CEC2015 learning-based optimization, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1098–1105

3. D. Aydın, T. Stutzle, A configurable generalized artificial bee colony algorithm with local search strategies, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1067–1074

4. O. Castillo, H. Neyoy, J. Soria, P. Melin, F. Valdez, A new approach for dynamic fuzzy logic parameter tuning in Ant Colony Optimization and its application in fuzzy control of a mobile robot. Appl. Soft Comput. **28**, 150–159 (2015)

5. O. Castillo, P. Melin, Intelligent adaptive model-based control of robotic dynamic systems with a hybrid fuzzy-neural approach. Appl. Soft Comput. **3**(4), 363–378 (2003)

6. Q. Chen, B. Liu, Q. Zhang, J.J. Liang, P.N. Suganthan, B.Y. Qu, Problem definition and evaluation criteria for CEC 2015 special session and competition on bound constrained single-objective computationally expensive numerical optimization. Computational Intelligence Laboratory, Zhengzhou University, China and Nanyang Technological University, Singapore, Technical report (2014)

7. L. Chen, C. Peng, H.L. Liu, S. Xie, An improved covariance matrix leaning and searching preference algorithm for solving CEC 2015 benchmark problems, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1041–1045

8. S.M. Guo, J.S.H. Tsai, C.C. Yang, P.H. Hsu, A self-optimization approach for L-SHADE incorporated with eigenvector-based crossover and successful-parent-selecting framework on CEC 2015 benchmark set. In *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1003–1010

9. X. Li, Decomposition and cooperative coevolution techniques for large scale global optimization. In *Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation* (ACM, New York, July 2014), pp. 819–838

10. J.J. Liang, L. Guo, R. Liu, B.Y. Qu, A self-adaptive dynamic particle swarm optimizer, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 3206–3213

11. R. Martínez-Soto, O. Castillo, L.T. Aguilar, Type-1 and Type-2 fuzzy logic controller design using a Hybrid PSO–GA optimization method. Inf. Sci. **285**, 35–49 (2014)

12. P. Melin, O. Castillo, A review on type-2 fuzzy logic applications in clustering, classification and pattern recognition. Appl. Soft Comput. **21**, 568–577 (2014)

13. P. Ochoa, O. Castillo, J. Soria, A fuzzy differential evolution method with dynamic adaptation of parameters for the optimization of fuzzy controllers, in *2014 IEEE Conference on Norbert Wiener in the 21st Century (21CW)* (IEEE, June 2014), pp. 1–6

14. P. Ochoa, O. Castillo, J. Soria, Differential evolution with dynamic adaptation of parameters for the optimization of fuzzy controllers, in *Recent Advances on Hybrid Approaches for designing intelligent systems* (Springer International Publishing, 2014), pp. 275–288

15. F. Olivas, F. Valdez, O. Castillo, P. Melin, Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. Soft. Comput. **20**(3), 1057–1070 (2016)

16. F. Olivas, O. Castillo, Particle swarm optimization with dynamic parameter adaptation using fuzzy logic for benchmark mathematical functions, in *Recent Advances on Hybrid Intelligent Systems* (Springer, Berlin, 2013), pp. 247–258

17. C. Peraza, F. Valdez, O. Castillo, A Harmony search algorithm comparison with genetic algorithms, in *Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics* (Springer International Publishing, 2015), pp. 105–123

18. C. Peraza, F. Valdez, O. Castillo, Fuzzy control of parameters to dynamically adapt the HS algorithm for optimization, in *Fuzzy Information Processing Society (NAFIPS) held jointly with 2015 5th World Conference on Soft Computing (WConSC), 2015 Annual Conference of the North American* (IEEE, New York, Aug 2015), pp. 1–6

19. R. Poláková, J. Tvrdík, P. Bujok, Cooperation of optimization algorithms: a simple hierarchical model, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1046–1052

20. K. Price, R.M. Storn, J.A. Lampinen, in *Differential Evolution: A Practical Approach to Global Optimization* (Springer Science & Business Media, 2006)
21. J.L. Rueda, I. Erlich, Testing MVMO on learning-based real-parameter single objective benchmark optimization problems, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, May 2015), pp. 1025–1032
22. K.M. Sallam, R.A. Sarker, D.L. Essam, S.M. Elsayed, Neurodynamic differential evolution algorithm and solving CEC2015 competition problems, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1033–1040
23. R. Storn, On the usage of differential evolution for function optimization, in *Fuzzy Information Processing Society, 1996. NAFIPS. 1996 Biennial Conference of the North American* (IEEE, New York, June 1996), pp. 519–523
24. R. Storn, K. Price, *Differential Evolution—A Simple and Efficient Adaptive Scheme for Global Optimization Over Continuous Spaces*, vol. 3 (ICSI, Berkeley, 1995)
25. R. Tanabe, A. Fukunaga, Success-history based parameter adaptation for differential evolution, in *2013 IEEE Congress on Evolutionary Computation* (IEEE, New York, June 2013), pp. 71–78
26. F. Valdez, P. Melin, O. Castillo, Evolutionary method combining particle swarm optimization and genetic algorithms using fuzzy logic for decision making, in *IEEE International Conference on Fuzzy Systems* (2009), pp. 2114–2119
27. F. Valdez, P. Melin, O. Castillo, Evolutionary method combining particle swarm optimisation and genetic algorithms using fuzzy logic for parameter adaptation and aggregation: the case neural network optimisation for face recognition. Int. J. Artif. Intell. Soft Comput. **2**(1–2), 77–102 (2010)
28. F. Valdez, P. Melin, O. Castillo, An improved evolutionary method with fuzzy logic for combining particle swarm optimization and genetic algorithms. Appl. Soft Comput. **11**(2), 2625–2632 (2011)
29. C. Yu, L.C. Kelley, Y. Tan, Dynamic search fireworks algorithm with covariance mutation for solving the CEC 2015 learning based competition problems, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1106–1112
30. Y.J. Zheng, X.B. Wu, Tuning maturity model of ecogeography-based optimization on CEC 2015 single-objective optimization test problems, in *2015 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, New York, May 2015), pp. 1018–1024