Soft Computing (2022) 26:12077–12092
https://doi.org/10.1007/s00500-022-07492-z

OPTIMIZATION

Gray wolf optimization-based self-organizing fuzzy multi-objective evolution algorithm

Jialiang Xie[1] · Shanli Zhang[1] · Honghui Wang[1] · Dongrui Wu[2,3]

Accepted: 8 April 2022 / Published online: 16 September 2022

Abstract

Two goals of multi-objective evolutionary algorithms are effectively improving their convergence and diversity and making the Pareto set evenly distributed and close to the real Pareto front. At present, the challenges to be solved by the multi-objective evolutionary algorithm are to improve the convergence and diversity of the algorithm, and how to better solve functions with complex PF and/or PS shapes. Therefore, this paper proposes a gray wolf optimization-based self-organizing fuzzy multi-objective evolutionary algorithm. Gray wolf optimization algorithm is used to optimize the initial weights of the self-organizing map network. New neighborhood relationships for individuals are built by self-organizing map, which can maintain the invariance of feature distribution and map the structural information of the current population into Pareto sets. Based on this neighborhood relationship, this paper uses the fuzzy differential evolution operator, which constructs a fuzzy inference system to dynamically adjust the weighting parameter in the differential operator, to generate a new initial solution, and the polynomial mutation operator to refine them. Boundary processing is then conducted. Experiments on 15 problems of GLT1-6 and WFG1-9 and the algorithm proposed in this paper achieve the best on 18 values. And the result shows that the convergence and diversity of the proposed algorithm are better than several state-of-the-art multi-objective evolutionary algorithms.

Keywords Gray wolf optimization · Fuzzy inference system · Self-organizing map · Evolutionary algorithm

1 Introduction

Many optimization problems in scientific research and industrial applications are intrinsic multi-objective, in which multiple conflicting objectives need to be optimized simultaneously. Thus, it is impossible to achieve the optimality of all problems at the same time. The solution is a Pareto optimal solution set (PS), consisting of multiple compromised solutions among different objectives. Vectors in the objective space that correspond to the PS are Pareto front (PF) (Zhou et al. 2009).

Common approaches to solve multi-objective optimization problems are traditional mathematical analytical algorithms and evolutionary algorithms. Multi-objective evolutionary algorithms (MOEAs) provide a general framework for solving complex problems and have been widely used in dynamic optimization, machine learning, signal processing, adaptive control, and so on. Popular MOEAs are usually based on Pareto dominance, performance indicator, and decomposition (Zhang et al. 2016).

An effective MOEA should make full use of the regularity property of multi-objective optimization problems (Zhang et al. 2016), that is, under certain conditions, the PF and PS of a continuous m-objective optimization problem form an (m-1)-dimensional piecewise continuous manifold in the objective space and the decision space, respectively. Regularity model-based multi-objective estimation of distribution algorithms (Zhang et al. 2008) explicitly uses this

✉ Jialiang Xie
xiejialiang@jmu.edu.cn

Shanli Zhang
845303782@qq.com

Honghui Wang
349638414@qq.com

Dongrui Wu
drwu@hust.edu.cn

[1] School of Science, Jimei University, Xiamen 361021, China

[2] School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China

[3] Key Laboratory of the Ministry of Education for Image Processing and Intelligent Control, Wuhan, China

property, for modeling the PS (Zhou et al. 2009) and for performing local search (Lara et al. 2010). Cellular multi-objective genetic algorithms (Durillo et al. 2008; Nebro et al. 2009; Zhang et al. 2015), MOEA based on decomposition (MOEAs/D) (Zhang et al. 2008; Li and Zhang 2009; Wang et al. 2016; Zhou and Zhang 2016), and hybrid NSGA-II with self-organizing map (Norouzi and Rakhshandehroo 2011) implicitly use this property.

However, in the evolutionary algorithm, there are uncertainties in the process of population search and the generation of offspring. Fuzzy set theory has inherent advantages in describing uncertain events and inaccurate information. Fuzzy inference system and hybrid methods with other intelligent computations are widely used in the field of evolutionary optimization, and have shown better results than traditional methods. For example, (Melin et al. 2013) used fuzzy logic to dynamically adjust the weight parameters C1 and C2 of the velocity formula in PSO. Olivas et al. (2017) proposed Ant Colony Optimization (ACO) with interval type-2 fuzzy system, which outperformed a rank-based ACO and ACO using type-1 fuzzy system. Santiago et al. (2019) proposed a novel MOEA with fuzzy logic-based adaptive selection of operators. It identifies which mutation operator is more (or less) promising among simulated binary crossover, uniform mutation, polynomial mutation, and DE, for the evolution of the population at each search stage. Also, it uses a fuzzy system to assign the correct application rate to these four operators. (Shen and Ge 2019 proposed a multi-objective particle swarm optimization algorithm based on fuzzy optimization, and the experiment has better performance in terms of solution quality, robustness, and computational complexity. HSMP (Zou et al. 2020) used the current and past continuous PS centers to automatically establish a T–S fuzzy nonlinear regression prediction model that can predict future PS centers to improve the prediction accuracy when environmental changes occur at the inflection point. Korashy et al. (2020)) proposed a method based on multi-objective gray wolf optimization and fuzzy logic decision-making for solving multi-robot coordination problems and a new objective function to minimize the recognition time between the main and backup relays. The feasibility and effectiveness of this method to solve the coordination problem of DOCRs were discussed on two different systems.

In addition, scholars merge fuzzy systems with machine learning and apply them to the field of multi-objective evolution to improve performance. For example, (Chen et al. 2018) proposed a hybrid population prediction strategy based on fuzzy inference and one-step prediction. A fuzzy inference model based on the maximum entropy principle is first extracted automatically from the previously found Pareto optimal solution set, and then the trajectory (position and/or direction) of the new Pareto optimal solution set is inferred. This strategy ensures that the algorithm can respond quickly and effectively when the environment changes. Changing PF, thereby, can be traced. Song et al. (2007) proposed a new fuzzy cognitive map (FCM) learning algorithm based on multi-objective particle swarm optimizations, and experimental results show that the method improves the efficiency and robustness of FCMs. In Yogesh and Ashish (2019), fuzzy logic was used to improve the adaptivity of particle swarm optimization (PSO) by controlling various parameters. Then, the improved PSO was used in K-harmonic means (KHM) for better clustering. Sankhwar et al. (2020) combined improved gray wolf optimization with fuzzy neural classifier for achieving more accurate financial crisis prediction than other methods.

The algorithms mentioned above cannot guarantee the diversity or convergence of the population well or cannot effectively solve functions with complex PF and/or PS shapes. In this paper, a gray wolf optimization-based self-organizing fuzzy multi-objective evolutionary algorithm (GWO-SFMEA) is proposed. Fuzzy system is used to dynamically adjust the weighted parameter F in differential evolution operator, and a new fuzzy differential evolution (FDE) operator is proposed. FDE is used to generate a new initial solution using neighborhood relationship among individuals, followed by polynomial mutation and boundary processing. The new neighborhood relationships of individuals are built by exploiting the peculiarity of the SOM (an unsupervised machine learning method), that is, invariance of the feature distribution, to map the structural information of the current population into Pareto sets. In addition, in this paper, gray wolf optimization is used to optimize the initial neuron weights of SOM.

Based on the above analysis, the main contributions of this paper are as follows:

1) Use GWO to optimize the weights of SOM, enabling individuals to search for their neighbors more efficiently in the global scope.
2) In order to produce high-quality new solutions, and improve the convergence and diversity of the algorithm. We use the fuzzy system to dynamically adjust the weighting parameter F in the process of offspring generation.
3) A gray wolf optimization-based self-organizing fuzzy multi-objective evolution algorithm is proposed to effectively solve multi-objective optimization problems.

Experiments on fifteen test problems with complex PF and/or PS shapes were conducted to verify the effectiveness of our proposed algorithm. Results show that its convergence and diversity are better than several state-of-the-art MOEAs.

The remainder of this paper is organized as follows: Sect. 2 reviews some preliminaries. Section 3 introduces the proposed GWO-SFMEA algorithm in detail. Section 4 presents the test instances and performance metrics. Section 5 describes the parameter settings and experimental results. Section 6 gives additional discussions about GWO-SFMEA. Finally, Sect. 7 draws conclusions.

2 Preliminaries

This paper considers the following form of multi-objective optimization problems (Marler and Arora 2004):

$$\begin{aligned} &\min \quad F(\boldsymbol{x}) = (f_1(\boldsymbol{x}),\ f_2(\boldsymbol{x}),\ \ldots,\ f_m(\boldsymbol{x})) \\ &s.t. \quad \boldsymbol{x} = (x_1,\ x_2,\ \ldots,\ x_n) \in \Omega \end{aligned} \tag{1}$$

where Ω is the feasible region of the decision space, $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \Omega$ is the decision variable vector. n is the dimensionality of $\boldsymbol{x}$, and m is the number of objective functions. $F : \Omega \to \mathbb{R}^m$ consists of m objective functions $\{f_i(\boldsymbol{x})\}_{i=1}^m$ from the decision space to the objective space.

In this section, we give some knowledge about stochastic operator and gray wolf optimization algorithm.

2.1 Stochastic operator

Most MOEAs are optimized by a set of candidate solutions to the target problems. These candidate solutions are generated by random operators. The main difference between MOEAs lies in the properties and search capabilities of the random operators. DE (Price et al. 2005) is one of the most effective operators to solve both single- and multi-objective continuous optimization problems (Li and Zhang 2009; Zhang et al. 2016; Ming et al. 2017; Bošković and Brest 2018).

Storn (1996), Mendes and Mohais (2005) proposed a variety of differential strategies to implement the mutation operation. Table 1 lists five of them. In $DE/\boldsymbol{x}/\boldsymbol{y}/\boldsymbol{z}$, $\boldsymbol{x}$ is a variation vector, which can be a random (rand) vector in the population, or the best (best) vector in the current population; $\boldsymbol{y}$ is the number of differential vectors; $\boldsymbol{z}$ represents the mode of crossover, and λ the combination factor.

$DE/rand/1/bin$ and $DE/best/2/bin$ are the most popular and successful differential strategies. This paper adopts the former to ensure the diversity of the population, and then integrates a fuzzy inference system into it.

Table 1 Differential strategies

Differential strategy	Differential expression
DE/rand/1/bin	$x_1 + F(x_2 - x_3)$
DE/rand/1/bin	$x_1 + F(x_2 + x_3 - x_4 - x_5)$
DE/best/1/bin	$x_b + F(x_2 - x_3)$
DE/best/2/bin	$x_b + F(x_2 + x_3 - x_4 - x_5)$
DE/rand-to-best/bin	$x_1 + \lambda(x_b - x_i) + F(x_2 - x_3)$

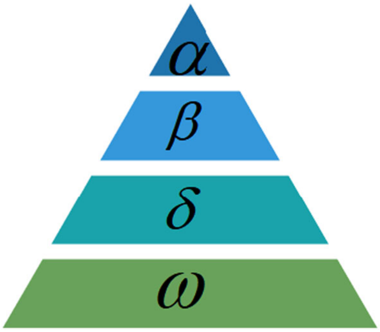


Fig. 1 Hierarchy of gray wolves

2.2 Gray wolf optimization (GWO)

GWO (Mirjalili et al. 2014; Saremi et al. 2015) is a new population intelligence optimization algorithm with fewer parameters, which is simple, flexible, and scalable. It has been widely used in many fields such as machine learning, image processing, and so on. For example, (Elhariri et al. 2015, 2016) successfully applied a GWO-based support vector machine to image classification and EMG signal classification. Mustaffa et al. (2015) used GWO to optimize the least square support vector machine, and applied it to commodity time series data.

In GWO, gray wolves strictly obey a social dominance hierarchy as shown in Fig. 1, where the α wolf is the leader of the population. The β wolf is a candidate for the α wolf, helping the α wolf make decisions or carry out other wolf group activities. The δ wolf complies with α wolf and β wolf, but dominates ω wolf.

In addition to the social hierarchy of wolves, GWO also includes tracking, encircling, attacking prey, etc.

A. Encircling prey

Its mathematical model is:

$$\overrightarrow{D} = |\overrightarrow{C} \cdot \overrightarrow{X_p(t)} - \overrightarrow{X}(t)| \tag{2}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X}_p(t) - \overrightarrow{A} \cdot \overrightarrow{D} \tag{3}$$

$$\overrightarrow{A} = 2\overrightarrow{a} \cdot \overrightarrow{r}_1 - \overrightarrow{a} \tag{4}$$

$$\overrightarrow{C} = 2 \cdot \overrightarrow{r}_2 \tag{5}$$

where t indicates the current iteration, $\overrightarrow{A}$ and $\overrightarrow{C}$ are coefficient vectors, $\overrightarrow{X}_p$ is the position vector of the prey, and $\overrightarrow{X}$ is the position of a gray wolf. In (4), components of $\overrightarrow{a}$ are linearly decreased from 2 to 0 over the course of iterations, and r_1 and r_2 are random vectors in $[0,\ 1]$.

B. Hunting

Its mathematical model is:

$$\begin{aligned} \overrightarrow{D}_\alpha &= |\overrightarrow{C}_1 \cdot \overrightarrow{X}_\alpha - \overrightarrow{X}|,\ \overrightarrow{D}_\beta \\ &= |\overrightarrow{C}_2 \cdot \overrightarrow{X}_\beta - \overrightarrow{X}|,\ \overrightarrow{D}_\delta = |\overrightarrow{C}_3 \cdot \overrightarrow{X}_\delta - \overrightarrow{X}| \end{aligned} \tag{6}$$

$$\overrightarrow{X}_1 = \overrightarrow{X}_\alpha - \overrightarrow{A}_1 \cdot \overrightarrow{D}_\alpha, \overrightarrow{X}_2 = \overrightarrow{X}_\beta - \overrightarrow{A}_2 \cdot \overrightarrow{D}_\beta, \overrightarrow{X}_3 = \overrightarrow{X}_\delta - \overrightarrow{A}_3 \cdot \overrightarrow{D}_\delta \tag{7}$$

$$\overrightarrow{X}(t+1) = \frac{\overrightarrow{X}_1 + \overrightarrow{X}_2 + \overrightarrow{X}_3}{3} \tag{8}$$

where $\overrightarrow{X}_\alpha$, $\overrightarrow{X}_\beta$ and $\overrightarrow{X}_\delta$ represent the position vectors of α, β and δ wolves in the current population, respectively. $\overrightarrow{D}_\alpha$, $\overrightarrow{D}_\beta$ and $\overrightarrow{D}_\delta$ indicate the distance between candidate wolves in the current population and α, β, and δ wolves, respectively.

C. Attacking prey (exploitation)

When the values of $\overrightarrow{A}$ are in $[-1, \ 1]$, the next position of the ω wolf which prepares for attacking the prey can be between its current position and the position of the prey. Otherwise, the wolves will spread out in search of prey for the sake of a global search and avoiding the local optimum.

3 GWO-based self-organizing fuzzy MOEA (GWO-SFMEA)

This section proposes a new gray wolf optimization-based self-organizing fuzzy multi-objective evolution algorithm. It first uses GWO to optimize the initial network weight vector of the SOM, and then SOM to extract the neighborhood relationship information of the best population individuals. In addition, it uses the fuzzy differential evolution (FDE) operator to generate a new solution.

3.1 GWO initialized SOM

SOM network (Kohonen 1990; Teuvo 1998), proposed by Kohonen et al, is an unsupervised machine learning method. The SOM generally consists of two parts: an input layer and an output layer. The neurons between the two layers are fully connected by a network weight vector. It adaptively adjusts this weight vector by detecting the relationship between the input data and the characteristic of the input data. The common topological structure of SOM is one- or two-dimensional. The topological structure of a two-dimensional SOM is shown in Fig. 2 (Zhang et al. 2016).

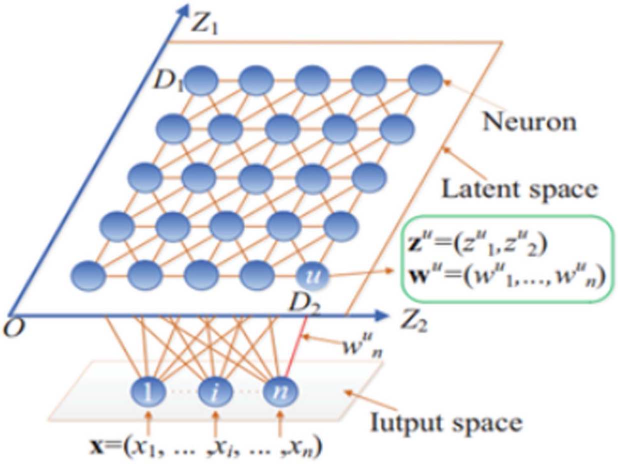

Z_1
D_1
Neuron
Latent space
$\mathbf{z}^u=(z^u{}_1,z^u{}_2)$
$\mathbf{w}^u=(w^u{}_1,\ldots,w^u{}_n)$
u
D_2
O
Z_2
$w^u{}_n$
1
i
n
Iutput space
$\mathbf{x}=(x_1, \ldots ,x_i, \ldots ,x_n)$

Fig. 2 An illustration of a two-dimensional SOM

To improve the performance of MOEAs, we utilize GWO to optimize the initial weight vector. Its fitness function is:

$$g(p_i) = \frac{100}{F(p_i)}, \tag{9}$$

where $F(p_i) = \sum_{n=1}^{N} \|x_q - c\|^2$ is used to measure the sum of the Euclidean distance between the input vector and the closest weight vector, and $c = arg \min_j \{\|x_q^T - \omega_j\|^2\}$. Algorithm 1 shows this optimization process.

Algorithm 1 GWO-SOM

```
1.Input: Population size of the gray wolves, boundary, number
  of neurons.
2. Set the maximum number of iterations about GWO, initialize
  gray wolf population (network weight).
3. Calculate the fitness by (9);
4. for $g = 1, 2, ..., maxIter$ do
5.   Boundary processing of the initial population.
6.   Choose $\alpha$, $\beta$, $\delta$ wolves and record the positions.
7.   Update the positions of other gray wolves according to (2)-(8).
8. end
9. Output: Positions of the population, i.e., the initial weight
  vector of SOM.
```

3.2 Fuzzy inference system

This paper uses Mamdani-type fuzzy system (Sivanandam et al. 2007) with triangular membership functions and centroid defuzzification to dynamically adjust the value of the mutation parameter F in the process of generating offspring using the DE operator. This mechanism allows different F values to be applied during each iteration so that the operator can generate higher-quality offspring. The exploration and development capabilities of the algorithm, therefore, can be improved.

During the generation of offsprings, the value of the weighted parameter F is generated by a fuzzy system. The fuzzy system first monitors the search process through the number of iterations and utilization, and then updates the value of the weighted parameter F according to the values of these two variables. The initial utilization value is 1. The corresponding utilization is decreased by $\frac{1}{\text{controlsize}}$ every time the F value is generated. The value of the $controlsize$ must be carefully tuned. It was set to 500 by trial and error.

In Figs. 3 and 4, we show the linguistic variables with the triangular membership functions for the number of iterations and utilization, respectively.

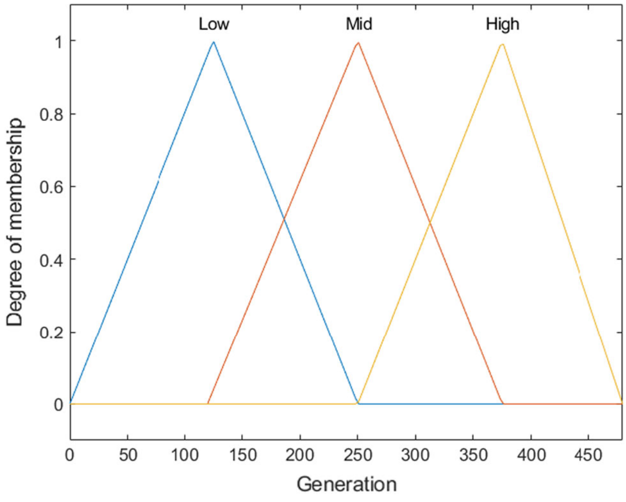


Fig. 3 Generation membership function and the representation of linguistic terms

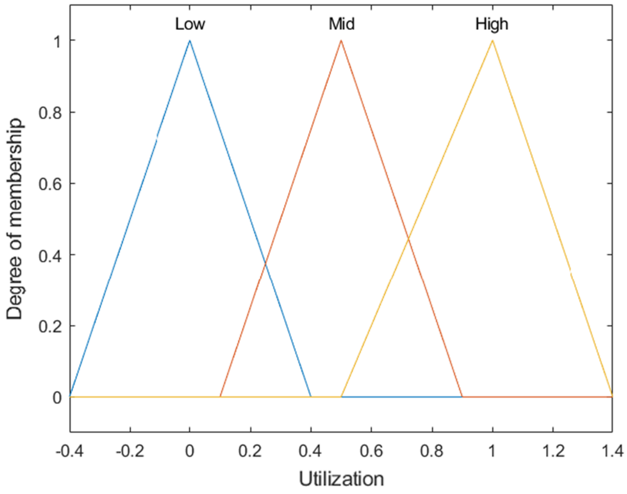


Fig. 4 Utilization membership function and the representation of linguistic terms

After the defuzzification process, we obtain a crisp value F. The specific surface function of fuzzy inference system is shown in Fig. 5, and its membership function as shown in Fig. 6.

To model the process with the fuzzy system, we consider 9 rules that help describe the existing relationship between the input and output, as shown in Table 2.

3.3 GWO-SFMEA framework

Figure 7 shows the specific process of the GWO-SFMEA algorithm.

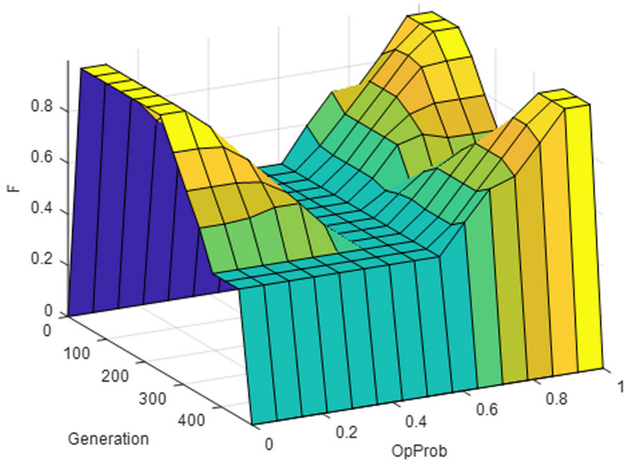


Fig. 5 Surface function graph of fuzzy inference system

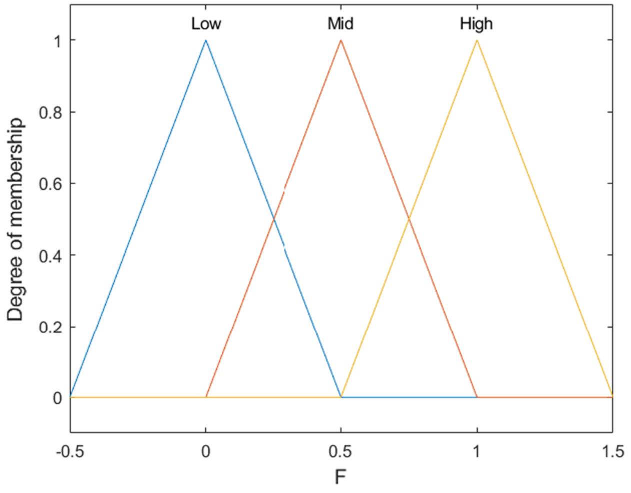


Fig. 6 The membership function and linguistic terms of output F

Table 2 Fuzzy inference rules

Fuzzy inference rules
R_1: if $Generation$ is low and $utilization$ is low, then F is high
R_2: if $Generation$ is low and $utilization$ is mid, then F is mid
R_3: if $Generation$ is low and $utilization$ is high, then F is high
R_4: if $Generation$ is mid and $utilization$ is low, then F is high
R_5: if $Generation$ is mid and $utilization$ is mid, then F is mid
R_6: if $Generation$ is mid and $utilization$ is high, then F is mid
R_7: if $Generation$ is high and $utilization$ is low, then F is high
R_8: if $Generation$ is high and $utilization$ is mid, then F is mid
R_9: if $Generation$ is high and $utilization$ is high, then F is high

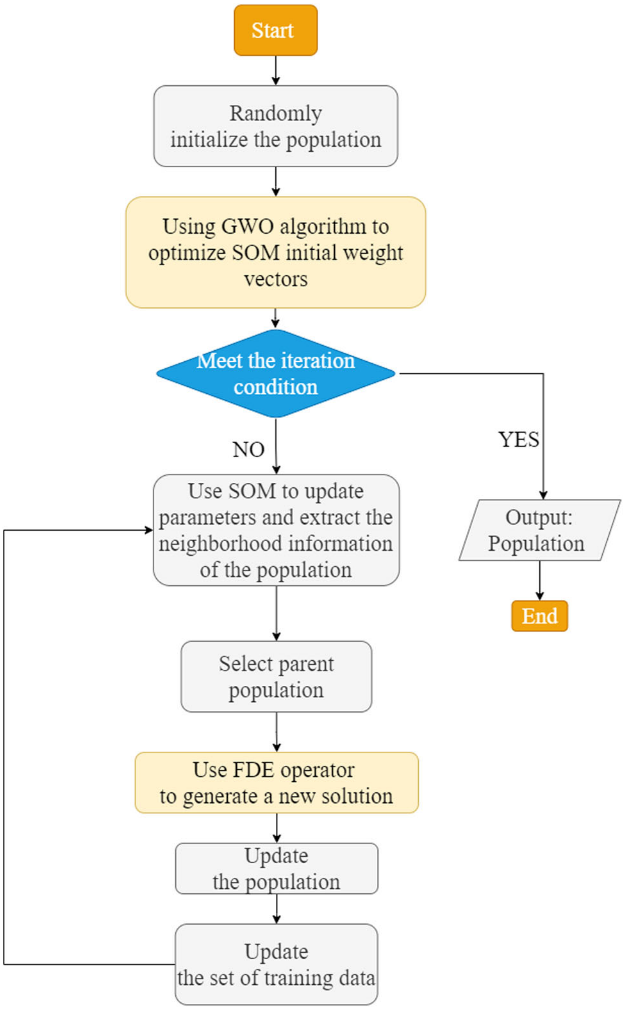

Start
Randomly initialize the population
Using GWO algorithm to optimize SOM initial weight vectors
Meet the iteration condition
NO
YES
Use SOM to update parameters and extract the neighborhood information of the population
Output: Population
End
Select parent population
Use FDE operator to generate a new solution
Update the population
Update the set of training data

Fig. 7 The GWO-SFMEA framework

(1) Randomly initialize the population and set parameters of the proposed algorithm;
(2) Use Algorithm 1 to initialize the weight vectors of the SOM;
(3) Start an iterative loop, use the training data set to update SOM network, i.e., the learning rate, weights, and neighborhood radius. And then extract neighborhood information between the population individuals;
(4) Use the tournament selection mechanism to select the parent population from neighbor or current population;
(5) Use the fuzzy system to dynamically adjust the weighting parameter F in each iteration.
(6) Use the FDE operator and the polynomial mutation operator to generate a new solution;
(7) Use the environmental selection mechanism in (Zhang et al. 2016) to update the population and training set. If the termination condition is met, output the optimal solution of the population; otherwise, repeat step (3).

4 Test instances and performance metrics

In this section, six MOPs with complex PF and PS shapes from (Gu et al. 2012; Zhang et al. 2016) are used as test functions. Table 3 introduces the feasible region of the set of test problems, the number of objective functions, and the dimensions of the decision variables.

In order to evaluate the convergence and diversity of the approximate PF obtained by the proposed algorithm, this paper adopts two commonly used performance indicators, i.e., $inverted\ generational\ distance(IGD)$ (Zhou et al. 2005; Zhang et al. 2008; Cai et al. 2020) and $hypervolume(HV)$ (Zitzler and Thiele 1999; Zhang et al. 2016).

Let PF^* and PF be the evenly distributed Pareto optimal solution set in PF and the obtained non-dominated front, respectively. IGD is computed by:

$$IGD(PF^*, PF) = \frac{\sum_{x^* \in PF^*} d(x^*, PF)}{|PF^*|},$$

where $d(x^*, PF)$ is the minimal distance between x^* and any point in PF, and $|PF^*|$ is the cardinality of PF^*. When using IGD metrics, the true PF value must be known. In the experiments of this section, 1000 uniformly distributed points are selected from PF to form $|PF^*|$.

Table 3 The GLT Test Instance Used in the Experiments

Instance	Variable space	Number of objective functions	Character
GLT1	$[0,1] \times [-1,1]^{n-1}$	2	Linear and disconnected PF; Nonlinear variable linkage
GLT2	$[0,1] \times [-1,1]^{n-1}$	2	Convex PF; Nonlinear variable linkage
GLT3	$[0,1] \times [-1,1]^{n-1}$	2	Convex PF; Nonlinear variable linkage
GLT4	$[0,1] \times [-1,1]^{n-1}$	2	Convex and disconnected PF; Nonlinear variable linkage
GLT5	$[0,1]^2 \times [-1,1]^{n-2}$	3	Convex PF; Nonlinear variable linkage
GLT6	$[0,1]^2 \times [-1,1]^{n-2}$	3	Convex and disconnected PF; Nonlinear variable linkage

The calculation of HV is:

$$HV(PF, r^*) = VOL\left(\bigcup_{x \in PF} [f_1(x), r_1^*] \times \cdots \times [f_m(x), r_m^*]\right),$$

where $r^* = (r_1^*, \ldots, r_m^*)$ is a reference point dominated by any Pareto optimal point in the objective space. $VOL(\cdot)$ is the Lebesgue measure. In our experiments, we set $r^* = (2, 2)$ for GLT1 and GLT3, $r^* = (2, 11)$ for GLT2, $r^* = (2, 3)$ for GLT4, $r^* = (2, 2, 2)$ for GLT5 and GLT6. Both the IGD and HV metrics measure the population convergence and diversity. The smaller (larger) the value of $IGD(HV)$ is, the better the performance of an algorithm.

5 Experimental results

In this section, the proposed algorithm GWO-SFMEA compares with SMEA (Zhang et al. 2016), MOEA/D-DE (Li and Zhang 2009), SOM-NSGA-II (Norouzi and Rakhshandehroo 2011), SMPSO (Liang et al. 2019) and FAME (Santiago et al. 2019) in the GLT test instances with regard to IGD value and HV value.

5.1 Parameter setting

The proposed algorithm is implemented on Windows 10, 64-bit, MATLAB2018a. The detailed parameters are set as follows:

- SOM structures: one-dimensional structure 1×100 for bi-objective MOPs, two-dimensional 7×15 for tri-objective MOPs; initial learning rate $\tau_0 = 0.7$;
- Size of neighborhood mating pools: $H = 5$;
- Probability of mating restriction: $\beta = 0.9$;
- Control parameters for FDE operator: $CR = 1$;
- Control parameters for PM: $p_m = \frac{1}{n}$, $\eta_m = 20$.

The implementation and parameter setting of other algorithms were configured according to the suggestions of the original papers. Table 4 summarizes these parameters.

5.2 Analysis of results

A. Time comparison

Table 5 shows the average time of SMEA and the proposed algorithm on 30 executions of 6 multi-objective test problems. Their time complexity on the bi-objective problem is similar, our proposed algorithm is faster on the tri-objective problem.

B. Comparison of IGD and HV value

The mean and standard deviation of the IGD and HV values of the 30 final populations generated by the six algorithms on the GLT test function are given in Table 6. The bold text indicates the best results (minimum IGD or maximum HV); “§,” “\” and “∼” in the table indicate that the performance of the algorithm GWO-SFMEA is better than, worse than, and similar to that of the comparison algorithm, respectively. Generally, the performance of GWO-SFMEA was significantly better than other algorithms on all GLT issues.

Table 4 Parameters settings

Algorithm	SOM-NSGA-II	SMEA
Population size	100/105 for bi/tri-objective MOPs	100/105 for bi/tri-objective MOPs
SOM structures	1×5/3×5 for bi/tri-objective MOPs	1×100/7×15 for bi/tri-objective MOPs
Differential evolution	F = 0.3, CR = 1.0	F = 0.9, CR = 1.0
Polynomial evolution	$p_m = 1/n$, $\eta_m = 20$	$p_m = 1/n$, $\eta_m = 20$
Neighborhood size		5, probability of mating restriction: 0.9
Algorithm	FAME	SMPSO
Population size	25/100 for bi/tri-objective MOPs	100/150 for bi/tri-objective MOPs
Archive size	100/200 for bi/tri-objective MOPs	100/150 for bi/tri-objective MOPs
SOM structures		1×100/15×10 for bi/tri-objective MOPs
Recombination	DE: F=0.5, CR=1.0, SBX: $\eta_c = 20$	
Polynomial evolution	$p_m = 0.3$, $\eta_m = 20$	
Uniform mutation	$p_m = 0.3$	
Algorithm	MOEA/D-DE	
Population size	100/105 for bi/tri-objective MOPs	
Differential evolution	F=0.9, CR=0.8	
Polynomial evolution	$p_m = 1/n$, $\eta_m = 20$	
Neighborhood size	5, probability of mating restriction: 0.9	

Table 5 Average time of SMEA and GWO-SFMEA on the GLT test suite/s

Instance	SMEA	GWO-SFMEA
GLT1	108.83	108.76
GLT2	104.47	98.67
GLT3	97.41	99.52
GLT4	97.72	99.19
GLT5	217.31	192.90
GLT6	214.44	204.19

Table 7 Average ranking of the algorithms

Algorithm	HV Rank	IGD Rank
SOM-NSGA-II	4.33	4.00
FAME	3.83	4.33
MOEA/D-DE	4.50	5.17
SMEA	2.67	2.00
SMPSO	4.17	4.50
GWO-SFMEA	1.50	1.00

In terms of IGD, the value achieved by the proposed algorithm was lower than the other five algorithms in both bi-objective and tri-objective problems. The performance of SMPSO on GLT1 and GLT3 was inferior to all algorithms. MOEA/D-DE performed the worst on GLT2, GLT5, and GLT6. The value obtained by FAME on GLT4 was inferior to other algorithms.

With regard to HV, the proposed algorithm obtained better values than other algorithms on all GLT test instances except GLT1 and GLT3, where FAME and SMPSO ranked first, respectively. However, FAME and MOEA/D-DE performed the worst on bi-objective problems(GLT4 for FAME, GLT2 and GLT3 for MOEA/D-DE). SMPSO had the worst performance on tri-objective problem(GLT5 and GLT6).

Table 7 shows the Friedman ranks of the six algorithms for the two considered indicators, with 95% significance. Regardless of IGD or HV, the average ranking of the proposed algorithm was the first, followed by SMEA. MOEA/D-

Table 6 Statistical result (Mean Std [Rank]) of the six algorithms on the GLT test suite of the IGD and HV metrics

Instance		SOM-NSGA-II	FAME	MOEA/D-DE	SMEA	SMPSO	GWO-SFMEA
IGD							
GLT1	Mean	4.437E–03§[4]	4.14E–03§[3]	5.522E–03§[5]	2.762E–03§[2]	2.38E–02§[6]	**2.107E–03[1]**
	Std	(2.239E–03)	(3.230E–03)	(2.681E–04)	(5.173E–04)	(2.25E–03)	(5.212E–05)
GLT2	Mean	4.345E–02§[4]	6.209E–02§[5]	3.579E–01§[6]	3.765E–02§[2]	3.82E–02§[3]	**3.699E–02[1]**
	Std	(3.080E–03)	(2.481E–02)	(5.802E–02)	(2.004E–03)	(1.88E–03)	(1.793E–03)
GLT3	Mean	8.704E–03§[3]	4.790E–02§[5]	2.348E–02§[4]	5.756E–03§[2]	9.32E–02§[6]	**5.165E–03[1]**
	Std	(4.219E–03)	(1.624E–02)	(3.311E–03)	(1.296E–03)	(5.32E–03)	(5.900E–04)
GLT4	Mean	1.769E–02§[3]	1.475E–01§[6]	2.162E–02§[4]	6.752E–03§[2]	4.17E–02§[5]	**5.876E–03[1]**
	Std	(2.702E–02)	(1.267E–01)	(3.928E–03)	(1.835E–02)	(4.90E–03)	(1.258E–04)
GLT5	Mean	5.530E–02§[5]	3.233E–02§[3]	8.278E–02§[6]	2.947E–02§[2]	4.77E–02§[4]	**2.933E–02[1]**
	Std	(3.843E–04)	(5.200E–04)	(2.006E–03)	(3.724E–04)	(2.57E–03)	(3.656E–04)
GLT6	Mean	5.123E–02§ [5]	4.810E–02§[4]	5.149E–02§[6]	2.163E–02§[2]	4.12E–02§[3]	**2.159E–02[1]**
	Std	(8.447E–03)	(5.266E–03)	(1.295E–03)	(3.489v04)	(3.34E–03)	(2.193E–04)
HV							
GLT1	Mean	3.357E+00§[5]	**3.377E+00\[1]**	3.369E+00∼[2]	3.366E+00§[4]	3.31E+00§[6]	3.369E+00[3]
	Std	(3.361E–03)	(9.401E–02)	(6.176-04)	(3.361E–03)	(5.30E–03)	(1.100E–03)
GLT2	Mean	1.975E+01§[5]	1.976E+01§[4]	1.942E+01§[6]	1.978E+01§[3]	1.98E+01∼[2]	**1.980E+01[1]**
	Std	(4.990E–03)	(1.504E–02)	(5.492E–02)	(5.825E–03)	(1.01E–02)	(2.032E–03)
GLT3	Mean	3.947E+00§[4]	3.939E+00§[6]	3.943E+00§[5]	3.948E+00§[3]	**3.95E+00∼[1]**	3.949E+00[2]
	Std	(1.024E–03)	(3.520E–03)	(6.799E–04)	(4.186E–04)	(1.06E–03)	(2.934E–04)
GLT4	Mean	4.979E+00§[3]	4.613E+00§[6]	4.969E+00§[5]	4.988E+00§[2]	4.97E+00§[4]	**4.992E+00[1]**
	Std	(2.934E–02)	(3.970E–01)	(8.920E–03)	(3.482E–02)	(2.59E–02)	(5.912E–04)
GLT5	Mean	7.947E+00§[4]	7.964E+00§[3]	7.940E+00§[5]	7.969E+00∼[2]	3.90E+00§[6]	**7.969E+00[1]**
	Std	(2.239E–03)	(5.000E–04)	(1.297E–03)	(1.617E–04)	(1.56E–02)	(9.977E–05)
GLT6	Mean	7.935E+00§[5]	7.945E+00§[3]	7.940E+00§[4]	7.962E+00∼[2]	3.86E+00§[6]	**7.962E+00[1]**
	Std	(6.048E–03)	(2.852E–02)	(3.323E–03)	(3.044E–04)	(1.37E–02)	(2.382E–04)

Bold indicates the value obtained by the first-ranked algorithm

DE ranked last on all metric values, which means it has the worst performance.

C. Population distribution graph

In order to further compare SMEA, FAME, and GWO-SFMEA, Fig. 8 shows the distribution of the final population implemented independently by three algorithms for 30 times.

For GLT1, the distribution of the final population achieved by FAME is better than SMEA and GWO-SFMEA, which is consistent with the optimal performance of HV of FAME in Table 6. Besides, the population distribution generated by GWO-SFMEA is better than SMEA. Both FAME and the proposed GWO-SFMEA apply fuzzy systems and outperform SMEA, indicating that the use of fuzzy systems in EAs can make the population distribution more uniform. The population distributions generated by SMEA, FAME, and GWO-SFMEA on GLT2 can cover the whole PFs well and do not show a big difference. SMEA is not much different from GWO-SFMEA on GLT3, but these two algorithms are better than FAME. The distribution of the population achieved by FAME, the right half part is not evenly distributed. Although the FAME guarantees the diversity of the population on GLT4, its distribution is uneven. By contrast, GWO-SFMEA guarantees uniformity. GWO-SFMEA performs better than the SEMA and FAME on tri-objective problems (GLT5 and GLT6).

By analyzing the above experimental results, we can found the SMPSO and MOEA/D-DE performed worst on both complicated PF or PS shapes, which might be because the particles have poor search ability or the set of the weight vectors. FAME works well on MOPs with three objectives, but it performs poorly on the GLT3 might be due to the random selection of the operator. SOM-NSGA-II and SMEA employ SOM to extract neighborhood information from the population, therefore, they can construct better quality parent population. Similarly, the algorithm proposed in this paper uses SOM to construct parent population, and uses the fuzzy system to dynamically adjust the parameters in the generation of its offspring to ensure that each iteration can produce high-quality solutions. Thus, GWO-SFMEA has a significant advantage in dealing with these instances.

6 Discussion

6.1 Test function with complex PF shapes

The WFG (Huband et al. 2005) test instances have a complex PF shape and a simple PS shape. Its characteristics are shown in Table 8.

Through preliminary experiments, the output surface of the fuzzy system with the mutation parameter F in the GWO-SFMEA algorithm is shown in Fig. 9, $\tau_0 = 0.9$, $CR = 0.8$ and the values of other parameters are the same as in Sect. 5.1.

The five algorithms were run on these instances 30 times. The mean and standard deviation of the IGD and HV values of the final population are shown in Table 9. The reference point for the bi-objective WFG that is used to calculate the HV metric is $r^* = (3, 5)$.

As shown in Table 9, the proposed algorithm achieved 8 best values on 18 average metric values. In terms of IGD, the proposed algorithm obtained lower values than other algorithms on WFG2, WFG5, WFG7, and WFG8. FAME performed best on the rest of the problems. MOEA/D-DE and SOM-NSGA-II obtained worst values on WFG1, WFG2, WFG6 and WFG3, WFG5, WFG9, respectively.

With regard to HV , the proposed algorithm was better than the other algorithms on WFG2, WFG5, WFG7-8. It was inferior to FAME and SMEA on WFG1, WFG4, WFG6, WFG9 and WFG3, respectively. However, FAME and SMEA obtain worst value on WFG8 and WFG1, respectively. MOEA /D-DE and SOM-NSGA-II obtained the worst values on WFG5-6 and WFG3, WFG7, WFG9, respectively.

In conclusion, GWO-SFMEA achieved good performance on WFG2, WFG5, WFG7, and WFG8 and ranked second on other problems. It can be seen that the algorithm proposed in this paper is able to tackle MOPs with complex PF shape.

Table 10 shows the Friedman ranks of the five algorithms for the two considered indicators, with 95% significance. The proposed algorithm achieved higher rank than other algorithms.

6.2 Application of fuzzy system

In order to study whether the fuzzy system improves the convergence of the algorithm and the diversity of the population, the GWO-SFMEA was compared with the version without fuzzy inference system by running two algorithms independently on the GLT problems 30 times. The average IGD and HV values after the implementation of two algorithms are shown in Fig. 10. The larger the HV value is or the smaller the IGD value is, the better performance of the algorithm is.

Observe Fig. 10a and compare the values in the figure. We can find that after using the fuzzy system, the IGD values of GLT1–GLT5 have all decreased, only the IGD of GLT6 has increased, but the increased accuracy is not large. Observing Fig. 10b, after using the fuzzy system, the HV values of all problems have increased.

In summary, the performance of the proposed algorithm on all instances was improved after adopting the fuzzy inference system, which shows that a well-organized fuzzy inference system in the MOEAs can improve the performance of the algorithm and can ensure the diversity of the population and the convergence speed of the population.

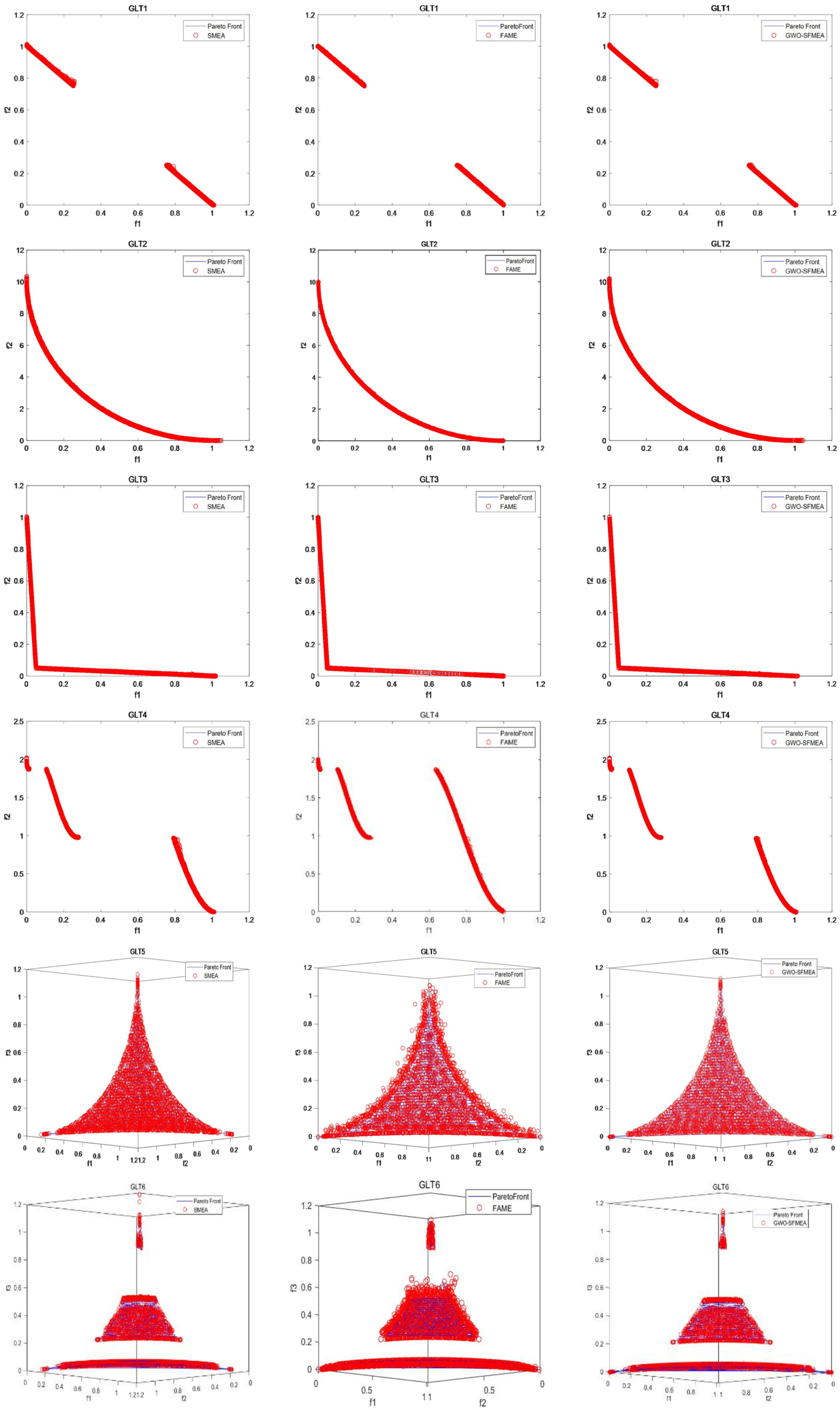


Fig. 8 The final populations in the objective space obtained by SMEA, FAME, and GWO-SFMEA on GLT1–GLT6 over 30 runs

Table 8 The WFG test instance used in the experiments

Instance	Character
WFG1	Separable; Uni-modal
WFG2	Non-separable; Uni-modal
WFG3	Non-separable; Uni-modal
WFG4	Separable; multi-modal
WFG5	Separable; deceptive
WFG6	Non-separable; Uni-modal
WFG7	Separate; Uni-modal
WFG8	Non-separable; Uni-modal
WFG9	Non-separable; multi-modal; deceptive

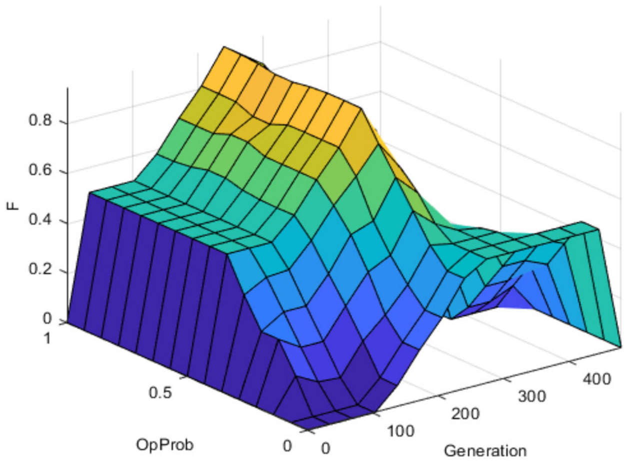


Fig. 9 Output surface of fuzzy system with weighted parameter F

6.3 Network structure of SOM

In GWO-SFMEA, the structure of SOM for the bi-objective and tri-objective problems is set to be one-dimensional (1 × 100) and two-dimensional (7 × 15), respectively. In order to study the influence of the SOM structure on the performance metrics IGD and HV of the algorithm, the SOM structure was changed to be two-dimensional (10 × 10) SOM in bi-objective problem and one-dimensional (1 × 105) in tri-objective problem. The two kinds of SOM structures were both run on GLT1–GLT6 for 30 times. The IGD and HV values obtained from each instance are compared with the unmodified ones. The results are shown in Fig. 11.

It can be observed from Fig. 11a that for the bi-objective instances, the IGD value on GLT1-2 is smaller under the one-dimensional SOM structure. However, the IGD value of GLT3-4 is smaller under the two-dimensional structure. For the tri-objective problem, the IGD value is smaller under the two-dimensional structure.

It can be seen from (b) that the change of the SOM structure has little effect on the HV value on all GLT problems. Strictly speaking, the HV value on the bi-objective problem is slightly larger under the one-dimensional structure, and that on the tri-objective problem is a little larger under the two-dimensional structure.

In conclusion, the bi-objective problem has no obvious preference for one-dimensional or two-dimensional SOM, but the tri-objective instance prefers two-dimensional SOM structure.

Table 9 Statistical Result (Mean Std [Rank]) of The Five Algorithms on the GLT Test Suite of the IGD and HV Metrics

Instance		SOM-NSGA-II	FAME	MOEA/D-DE	SMEA	GWO-SFMEA
IGD						
WFG1	Mean	1.433E+00§[3]	**1.491E–02\[1]**	1.695E+00§[5]	1.565E+00§[4]	1.085E+00[2]
	Std	(7.333E–02)	(1.308E–03)	(1.122E–02)	(9.525E–02)	(2.019E–02)
WFG2	Mean	1.433E–02§[2]	1.846E–02§[4]	3.790E–02§[5]	1.474E–02§[3]	**1.121E–02[1]**
	Std	(6.093E–03)	(3.859E–02)	(6.109E–04)	(7.078E–04)	(4.670E–04)
WFG3	Mean	1.450E–01§[5]	**1.127E–02\[1]**	1.420E–01§[4]	1.388E–01§[3]	1.137E–02[2]
	Std	(6.990E–04)	(6.682E–05)	(3.025E–04)	(1.380E–04)	(7.061E–05)
WFG4	Mean	6.037E–02§[4]	**1.084E–02\[1]**	5.025E–02§[3]	7.784E–02§[5]	3.059E–02[2]
	Std	(5.681E–03)	(1.364E–04)	(6.321E–03)	(6.902E–03)	(3.959E–02)
WFG5	Mean	6.902E–02§[5]	6.638E–02§[2]	6.747E–02§[4]	6.663E–02§[3]	**6.467E–02[1]**
	Std	(5.246E–04)	(4.037E–05)	(4.476E–04)	(1.138E–04)	(5.701E–03)
WFG6	Mean	3.334E–01§[4]	**1.288E–02\[1]**	3.352E–01§[5]	3.265E–01§[3]	3.022E–01[2]
	Std	(4.137E–03)	(2.792E–04)	(2.240E–02)	(2.090E–02)	(3.069E–04)
WFG7	Mean	2.994E–02§[5]	1.261E–02§[3]	1.815E–02§[4]	1.144E–02§[2]	**1.071E–02[1]**
	Std	(1.527E–03)	(2.062E–04)	(2.968E–04)	(2.526E–04)	(3.107E–04)
WFG8	Mean	5.755E–02§[4]	1.033E–01§[5]	3.989E–02§[3]	2.666E–02§[2]	**1.186E–02[1]**
	Std	(6.643E–03)	(2.750E–02)	(7.611E–03)	(6.924E–03)	(2.819E–04)

Table 9 continued

Instance		SOM-NSGA-II	FAME	MOEA/D-DE	SMEA	GWO-SFMEA
IGD						
WFG9	Mean	2.707E–01§[5]	**1.448E–02\[1]**	2.435E–01§[3]	2.561E–01§[4]	1.984E–01[2]
	Std	(1.217E–02)	(4.970E–03)	(3.200E–02)	(2.980E–02)	(7.404E–04)
HV						
WFG1	Mean	4.428E+00§[4]	**1.207E+01\[1]**	5.809E+00§[3]	3.873E+00§[5]	5.878E+00[2]
	Std	(2.667E–01)	(4.180E–03)	(5.017-02)	(3.781E–01)	(1.217E–01)
WFG2	Mean	1.141E+01§[3]	1.136E+01§[5]	1.143E+01§[2]	1.141E+01§[4]	**1.145E+01[1]**
	Std	(3.455E–03)	(4.195E–01)	(1.397E–03)	(6.347E–03)	(5.476E–02)
WFG3	Mean	1.089E+01§[5]	1.096E+01∼[2]	1.091E+01§[4]	**1.594E+01\[1]**	1.096E+01[3]
	Std	(5.643E–03)	(2.100E–04)	(3.319E–03)	(1.946E–03)	(1.141E–03)
WFG4	Mean	8.328E+00§[4]	**8.684E+00\[1]**	8.443E+00§[3]	8.204E+00§[5]	8.559E+00[2]
	std	(3.591E–02)	(1.710E–03)	(3.884E–02)	(3.966E–02)	(1.993E–02)
WFG5	Mean	8.158E+00§[4]	8.181E+00§[3]	8.138E+00§[5]	8.233E+00§[2]	**8.251E+00[1]**
	Std	(5.498E–02)	(3.902E–02)	(3.069E–02)	(4.557E–02)	(3.188E–02)
WFG6	Mean	6.344E+00§[4]	**8.677E+00\[1]**	6.337E+00§[5]	6.383E+00§[3]	6.527E+00[2]
	Std	(2.199E–03)	(3.651E–02)	(1.139E–01)	(1.064E–01)	(3.112E–04)
WFG7	Mean	8.575E+00§[5]	8.686E+00§[2]	8.644E+00§[4]	8.671E+00§[3]	**8.687E+00[1]**
	Std	(7.901E–03)	(2.620E–03)	(3.616E–03)	(2.129E–03)	(2.600E–04)
WFG8	Mean	8.352E+00§[4]	7.049E+00§[5]	8.482E+00§[3]	8.528E+002§[2]	**8.675E+00[1]**
	Std	(3.872E–02)	(4.843E–01)	(4.711E–02)	(4.456E–02)	(2.495E–03)
WFG9	Mean	6.102E+00§[5]	**8.448E+00\[1]**	6.179E+00§[3]	6.174E+00§[4]	6.451E+00[2]
	Std	(6.438E–02)	(4.316E–02)	(1.563E–01)	(1.518E–01)	(2.543E–02)

Bold indicates the value obtained by the first-ranked algorithm

Table 10 Average rankings of the algorithms

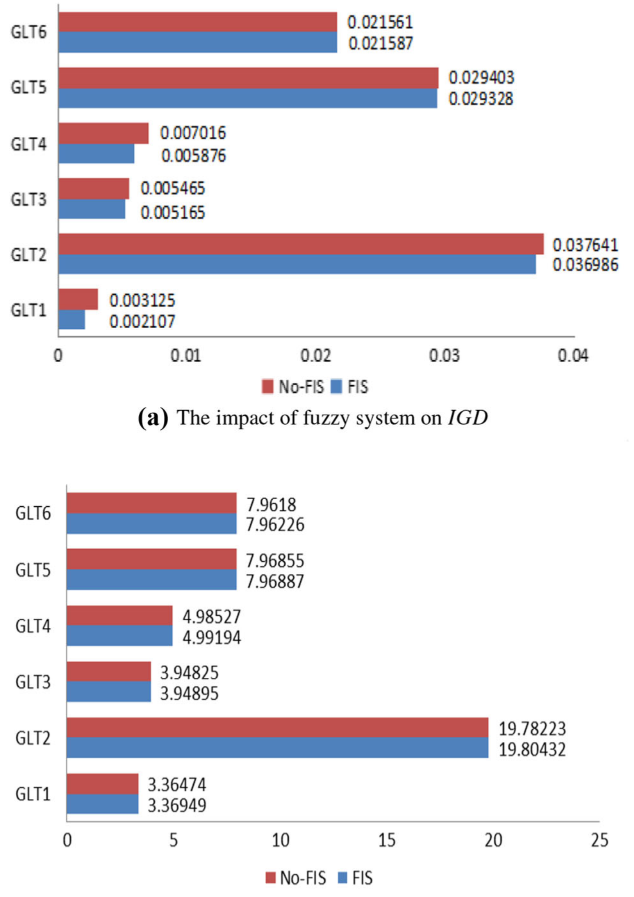
Algorithm	HV rank	IGD rank
SOM-NSGA-II	4.22	4.11
FAME	2.33	2.11
MOEA/D-DE	3.56	4.00
SMEA	3.22	3.22
GWO-SFMEA	1.67	1.56

6.4 Comparison of GWO and other optimization algorithms

In this paper, GWO is used to optimize the initial weight of SOM. In order to further illustrate the advantages of using GWO, we compare GWO with the latest optimization algorithms—bald eagle algorithm(BES)(Alsattar et al. 2020) and slime mold algorithm(SMA)(Li et al. 2020). BES is a new meta-heuristic algorithm proposed by Alsattar in 2020. (Angayarkanni et al. 2021) combined BES algorithm with the gray wolf optimization algorithm to optimize the parameters of support vector regression in the prediction of traffic flow. The experiment shows its effective application in intelligent transportation. Xie et al. (2021) proposed a network public opinion prediction model based on the bald eagle algorithm optimized radial basis function neural network (BES-RBF). This model can better describe the development trend of different network public opinion information. SMA (Li et al. 2020) is another new meta-heuristic algorithm proposed by Li in 2020. Mostafa et al. (2020) used SMA to extract the global optimal values of solar photovoltaic cell parameters, and SMA can handle the nonlinearity and multi-modal properties of photovoltaic cell characteristics.

In this subsection, the above three algorithms are all used to optimize the initial weights of SOM and test on GLT1-6. In order to show the advantages and disadvantages of the three algorithms more directly, no fuzzy system is added in this experimental part. The mean and standard deviation of the IGD and HV values of the 30 final populations generated by the three algorithms on the GLT test function are given in Table 11. The gray background indicates the best results (minimum IGD or maximum HV); "§," "\" and "∼" in the table indicate that the performance of the algorithm GWO-SOM is better than, worse than, and similar to that of the comparison algorithm, respectively.

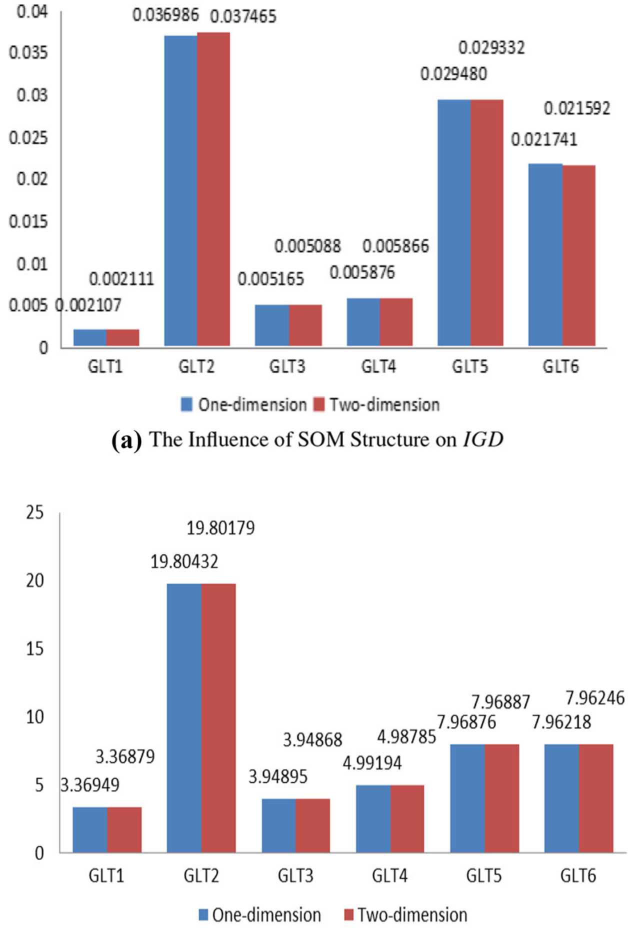
Observing the above table, among the above 12 values, GWO, BES and SMA dominate by 7, 0, and 5, respectively. In terms of IGD, GWO is used to optimize the initial weight of SOM, and the IGD value obtained is the best on GLT1, 5, and 6. Using SMA to optimize the initial weight of SOM is



(a) The impact of fuzzy system on *IGD*

(b) The impact of fuzzy system on *HV*

Fig. 10 The impact of fuzzy system on IGD and HV



(a) The Influence of SOM Structure on *IGD*

(b) The Influence of SOM Structure on *HV*

Fig. 11 The Influence of SOM Structure on IGD and HV

the best on other problems. With regard to HV, the method used in this paper achieves the best value on GLT2, GLT4-6, and SMA-SOM achieves the best value on GLT1 and GLT3. On tri-objective problems, the average values obtained by the three algorithms are the same, but the standard deviation obtained by GWO-SOM is the smallest, indicating that the HV value obtained by GWO-SOM in running 30 times is stable. Therefore, this paper uses GWO to optimize the initial weight of SOM, and compared with BES and SMA, the algorithm has no redundant parameters, does not increase the time complexity of the algorithm, and is easy to operate.

7 Conclusions

In this paper, a new MOEA called gray wolf optimization-based self-organizing fuzzy multi-objective evolution algorithm was proposed. In algorithms which combine SOM with MOEAs, the initial weight of self-organizing map is generated randomly. However, the result depends heavily on the initial weight. Therefore, this paper optimized the initial weights of SOM through gray wolf optimization algorithm. At the same time, SOM can be used to map high-dimensional information to low-dimensional space while maintaining feature distribution invariance and building neighborhood relationships between individuals. In addition, the suboptimal solutions produced in the early optimization process in most MOEAs induce the algorithm to converge prematurely with great probability. To avoid this problem, this paper first utilized FDE operator to generate a new initial solution. Then the PM operator was used to mutate the new solution and boundary processing was performed. FDE dynamically adjusts the weighting parameter F in the difference operator by constructing a fuzzy inference system, which can ensure high-quality new solutions during the generation of offspring and effectively improve the convergence and diversity of the algorithm.

In order to verify the performance of the proposed algorithm, GWO-SFMEA and other state-of-the-art algorithms were tested on the GLT and WFG test functions with complex PF shapes and/or complex PS shapes, respectively.

Table 11 Statistical result (Mean Std [Rank]) of the three algorithms on the GLT test suite of the IGD and HV metrics

Instance		GWO-SOM	BES-SOM	SMA-SOM
IGD				
GLT1	Mean	**3.013E–03[1]**	3.610E-03§[3]	3.039E-03§[2]
	Std	(5.475E–03)	(3.267E–03)	(4.250E–04)
GLT2	Mean	3.798E–02[3]	3.762E–02\[2]	**3.746E–02\[1]**
	Std	(1.741E–03)	(1.591E–03)	(2.240E–03)
GLT3	Mean	6.503E–03[2]	6.575E–03§[3]	**5.311E–03\[1]**
	Std	(4.354E–03)	(6.061E–03)	(2.209E–04)
GLT4	Mean	5.990E–03[2]	6.109E–03§[3]	**5.738E–03\[1]**
	Std	(2.342E–03)	(2.184E–03)	(2.479E–03)
GLT5	Mean	**2.960E–02[1]**	2.964E–02§[2]	2.969E–02§[3]
	Std	(3.657E–04)	(3.429E–04)	(3.734E–04)
GLT6	Mean	**2.172E–02[1]**	2.183E–02§[3]	2.180E–02§[2]
	Std	(3.143E–04)	(3.121E–04)	(3.476E–04)
HV				
GLT1	Mean	3.364E+00[2]	3.362E+00§[3]	**3.369E+00\[1]**
	Std	(4.576E–03)	(1.222E–01)	(3.502-03)
GLT2	Mean	**1.978E+01[1]**	1.977E+01§[3]	1.978E+01§[2]
	Std	(6.293E–03)	(6.217E–03)	(8.306E–04)
GLT3	Mean	3.948E+00[2]	3.948E+00§[3]	**3.948E+00\[1]**
	Std	(9.141E–04)	(1.238E–03)	(1.692E–04)
GLT4	Mean	**4.986E+00[1]**	4.974E+00§[3]	4.984E+00§[2]
	Std	(5.130E–03)	(3.249E–02)	(1.369E–02)
GLT5	Mean	**7.968E+00[1]**	7.968E+00§[2]	7.968E+00§[3]
	Std	(1.435E–04)	(1.485E–04)	(1.573E–04)
GLT6	Mean	**7.961E+00[1]**	7.961E+00§[2]	7.961E+00§[3]
	Std	(3.197E–04)	(3.757E–04)	(3.469E–04)

Bold indicates the value obtained by the first-ranked algorithm

Through experiments, it can be found that the proposed algorithm is significantly better than other algorithms at the 95% confidence level. Through analysis of the sensitivity of GWO-SFMEA to the SOM structure, it can be seen that the SOM structure has no obvious preference for the bi-objective problem, but the tri-objective problem prefers the two-dimensional SOM structure. In addition, the combination of fuzzy systems with EAs can greatly improve the convergence and diversity of evolutionary algorithms.

The fuzzy rules designed in this paper are obtained through repeated trial and error and expert knowledge, which will affect the performance of the algorithm to a certain extent. Therefore, in future research, a fuzzy rule generator will be explored, which can generate fuzzy rules suitable for solving the problem according to the characteristics of the problem to be optimized. Moreover, we will look for different ways to improve the performance of the algorithm so that it can effectively deal with mixed and separable problems.

Author Contributions Jialiang Xie gave the overall framework of the paper and introduction, and Shanli Zhang completed the experiment and the first draft. Honghui Wang modified the grammar of the paper, and Dongrui Wu coordinated the paper.

Funding This work was supported in part by the National Natural Science Foundation of China (Nos. 12271211,12071179), the National Natural Science Foundation of Fujian Province (Nos. 2021J01861), Soft Science Research Program of Fujian Province (No. B19085), the Project of Education Department of Fujian Province (No. JT180263), the Youth Innovation Fund of Xiamen City (3502Z20206020), the Open Fund of Digital Fujian Big Data Modeling and Intelligent Computing Institute, Pre-Research Fund of Jimei University.

Data availability Enquiries about data availability should be directed to the authors.

Declarations

Conflict of interest All authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Alsattar H, Zaidan A, Zaidan B (2020) Novel meta-heuristic bald eagle search optimisation algorithm. Artif Intell Rev 53(6):2237–2264

Angayarkanni S, Sivakumar R, Rao Y (2021) Hybrid grey wolf: bald eagle search optimized support vector regression for traffic flow forecasting. J Ambient Intell Human Comput 12(1):1293–1304

Bošković B, Brest J (2018) Protein folding optimization using differential evolution extended with local search and component reinitialization. Inf Sci 454:178–199

Cai XY, Xiao YS, Li MQ, Hu H, Ishibuchi H, Li XP (2020) A grid-based inverted generational distance for multi/many-objective optimization. IEEE Trans Evol Comput 25(1):21–34

Chen DB, Zou F, Lu RQ, Wang XD (2018) A hybrid fuzzy inference prediction strategy for dynamic multi-objective optimization. Swarm Evol Comput 43:147–165

Durillo JJ, Nebro AJ, Luna F, Alba E (2008) Solving three-objective optimization problems using a new hybrid cellular genetic algorithm. International conference on parallel problem solving from nature. Springer, Berlin, pp 661–670

Elhariri E, El-Bendary N, Hassanien AE, Abraham A (2015) Grey wolf optimization for one-against-one multi-class support vector machines. In: 2015 7th international conference of soft computing and pattern recognition (SoCPaR). IEEE, pp 7–12

Elhariri E, El-Bendary N, Hassanien AE (2016) A hybrid classification model for EMG signals using grey wolf optimizer and SVMs. In: The 1st international conference on advanced intelligent system and informatics (AISI2015), November 28–30, 2015. Springer, Beni Suef, Egypt, pp 297–307

Gu FQ, Liu HL, Tan KC (2012) A multiobjective evolutionary algorithm using dynamic weight design method. Int J Innov Comput Inf Control 8(5B):3677–3688

Huband S, Barone L, While L, Hingston P (2005) A scalable multi-objective test problem toolkit. In: International conference on evolutionary multi-criterion optimization. Springer, pp 280–295

Kohonen T (1990) The self-organizing map. Proc IEEE 78(9):1464–1480

Korashy A, Kamel S, Nasrat L, Jurado F (2020) Developed multi-objective grey wolf optimizer with fuzzy logic decision-making tool for direction overcurrent relays coordination. Soft Comput 24(17):13305–13317

Lara A, Sanchez G, Coello CAC, Schutze O (2010) HCS: a new local search strategy for memetic multiobjective evolutionary algorithms. IEEE Trans Evol Comput 14(1):112–132

Li H, Zhang QF (2009) Multiobjective optimization problems with complicated pareto sets, MOEA/D and NSGA-II. IEEE Trans Evol Comput 13(2):284–302

Li SM, Chen HL, Wang MJ, Heidari AA, Mirjalili S (2020) Slime mould algorithm: a new method for stochastic optimization. Futur Gener Comput Syst 111:300–323

Liang J, Guo QQ, Yue CT, Qu BY (2019) Self-organizing multi-objective particle swarm optimization algorithm. Appl Res Comput 36(8):2311–2316

Marler RT, Arora JS (2004) Survey of multi-objective optimization methods for engineering. Struct Multidiscip Optim 26(6):369–395

Melin P, Olivas F, Castillo O, Valdez F, Soria J, Valdez M (2013) Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic. Expert Syst Appl 40(8):3196–3206

Mendes R, Mohais AS (2005) DynDE: a differential evolution for dynamic optimization problems. 2005 IEEE congress on evolutionary computation 3:2808–2815

Ming MJ, Wang R, Zha YB, Zhang T (2017) Pareto adaptive penalty-based boundary intersection method for multi-objective optimization. Inf Sci 414:158–174

Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69(3):46–61

Mostafa M, Rezk H, Aly M, Ahmed EM (2020) A new strategy based on slime mould algorithm to extract the optimal model parameters of solar pv panel. Sustain Energy Technol Assess 42:100849–100861

Mustaffa Z, Sulaiman MH, Kahar MNM (2015) Training LSSVM with GWO for price forecasting. In: 2015 international conference on informatics. Electronics and vision (ICIEV). IEEE, pp 1–6

Nebro AJ, Durillo JJ, Luna F, Dorronsoro B, Alba E (2009) MOCell: a cellular genetic algorithm for multiobjective optimization. Int J Intell Syst 24(7):726–746

Norouzi K, Rakhshandehroo GR (2011) A self organizing map based hybrid multi-objective optimization of water distribution networks. Iran J Sci Technol Trans B Eng 35(1):105–119

Olivas F, Valdez F, Castillo O, Gonzalez CI, Martinez G, Melin P (2017) Ant colony optimization with dynamic parameter adaptation based on interval type-2 fuzzy logic systems. Appl Soft Comput 53:74–87

Price KV, Storn RM, Lampinen JA (2005) Differential evolution-A practical approach to global optimization. Springer, Berlin

Sankhwar S, Gupta D, Ramya KC, Rani SS, Shankar K, Lakshmanaprabu SK (2020) Improved grey wolf optimization-based feature subset selection with fuzzy neural classifier for financial crisis prediction. Soft Comput 24(1):101–110

Santiago A, Dorronsoro B, Nebro AJ, Durillo JJ, Fraire HJ (2019) A novel multi-objective evolutionary algorithm with fuzzy logic based adaptive selection of operators: FAME. Inf Sci 471:233–251

Saremi S, Mirjalili SZ, Mirjalili SM (2015) Evolutionary population dynamics and grey wolf optimizer. Neural Comput Appl 26(5):1257–1263

Shen YP, Ge GR (2019) Multi-objective particle swarm optimization based on fuzzy optimality. IEEE Access 7:101513–101526

Sivanandam N, Sai S, Deepa SN (2007) Introduction to fuzzy logic using MATLAB. Springer, Berlin

Song HJ, Miao CY, Shen ZQ (2007) Fuzzy cognitive map learning based on multi-objectiveparticle swarm optimization. In: Proceedings of the 9th annual conference on Genetic and evolutionary computation, pp 339–339

Storn R (1996) On the usage of differential evolution for function optimization. In: Fuzzy information processing society, Nafips Biennial Conference of the North American, pp 519–523

Teuvo K (1998) The self-organizing map. Neurocomputing 21(1):1–6

Wang LP, Zhang QF, Min ZA, Gong MG, Jiao LC (2016) Constrained subproblems in decomposition based multiobjective evolutionary algorithm. IEEE Trans Evol Comput 20(3):475–480

Xie JL, Zhang SL, Lin L (2021) Prediction of network public opinion based on bald eagle algorithm optimized radial basis function neural network. Int J Intell Comput Cybern. https://doi.org/10.1108/IJICC-07-2021-0148

Yogesh G, Ashish S (2019) A new swarm-based efficient data clustering approach using KHM and fuzzy logic. Soft Comput 23(1):145–162

Zhang H, Song SM, Zhou AM, Gao XZ (2015) A multiobjective cellular genetic algorithm based on 3D structure and cosine crowding measurement. Int J Mach Learn Cybern 6(3):487–500

Zhang H, Zhou AM, Song SM, Zhang QF, Gao XZ, Zhang J (2016) A self-organizing multiobjective evolutionary algorithm. IEEE Trans Evol Comput 20(5):792–806

Zhang QF, Zhou AM, Jin YC (2008) RM-MEDA: a regularity model-based multiobjective estimation of distribution algorithm. IEEE Trans Evol Comput 12(1):41–63

Zhou AM, Zhang QF (2016) Are all the subproblems equally important? Resource allocation in decomposition-based multiobjective evolutionary algorithms. IEEE Trans Evol Comput 20(1):52–64

Zhou AM, Zhang QF, Jin YC, Tsang E (2005) A model-based evolutionary algorithm for bi-objective optimization. In: 2005 IEEE congress on evolutionary computation, vol 3, pp 2568–2575

Zhou AM, Zhang QF, Jin YC (2009) Approximating the set of pareto-optimal solutions in both the decision and objective spaces by an estimation of distribution algorithm. IEEE Trans Evol Comput 13(5):1167–1189

Zitzler E, Thiele L (1999) Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Trans Evol Comput 3(4):257–271

Zou F, Chen DB, Xu QZ, Lu RQ (2020) A new prediction strategy combining T-S fuzzy nonlinear regression prediction and multi-step prediction for dynamic multi-objective optimization. Swarm Evol Comput 59:100749–100768

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.