

# A Co-optimization PSO for Fuzzy Rule-Based Classifier Design Problem Based on Enlarged Hedge Algebras

Du Duc Nguyen<sup>1</sup>, Phong Dinh Pham<sup>2\*</sup>

<sup>1</sup> Software Engineering Department, Faculty of Information Technology, University of Transport and Communications, 3 Cau Giay street, Lang Thuong ward, Dong Da District, 100000 Hanoi, Vietnam

<sup>2</sup> Computer Science Department, Faculty of Information Technology, University of Transport and Communications, 3 Cau Giay street, Lang Thuong ward, Dong Da District, 100000 Hanoi, Vietnam

\* Corresponding author, e-mail: [phongpd@utc.edu.vn](mailto:phongpd@utc.edu.vn)

Received: 08 April 2020, Accepted: 04 December 2020, Published online: 14 October 2021

## Abstract

Fuzzy Rule-Based Classifier (FRBC) design problem has been widely studied due to many practical applications. Hedge Algebras based Classifier Design Methods (HACDMs) are the outstanding and effective approaches because these approaches based on a mathematical formalism allowing the fuzzy sets based computational semantics generated from their inherent qualitative semantics of linguistic terms. HACDMs include two phase optimization process. The first phase is to optimize the semantic parameter values by applying an optimization algorithm. Then, in the second phase, the optimal fuzzy rule based system for FRBC is extracted based on the optimal semantic parameter values provided by the first phase. The performance of FRBC design methods depends on the quality of the applied optimization algorithms. This paper presents our proposed co-optimization Particle Swarm Optimization (PSO) algorithm for designing FRBC with trapezoidal fuzzy sets based computational semantics generated by Enlarged Hedge Algebras (EHAs). The results of experiments executed over 23 real world datasets have shown that Enlarged Hedge Algebras based classifier with our proposed co-optimization PSO algorithm outperforms the existing classifiers which are designed based on Enlarged Hedge Algebras methodology with two phase optimization process and the existing fuzzy set theory based classifiers.

## Keywords

Enlarged Hedge Algebras (EHAs), co-optimization algorithm, PSO, Fuzzy Rule-Based Classifier (FRBC)

## 1 Introduction

Fuzzy Rule-Based Classifiers (FRBCs) have many applications in the field of data mining. The advantage of FRBCs is that end-users can exploit the highly interpretable classification models in the form of if-then fuzzy rules which are extracted from data after a one-time training process as their knowledge.

Fuzzy Rule-Based Classifier (FRBC) design methods which utilize fuzzy set theory extract the fuzzy classification rules for classifiers from the pre-designed fuzzy partitions using the fuzzy sets which linguistic terms of linguistic variables are assigned to them by human experts [1–6]. So, the linguistic terms are just the linguistic labels assigned to the fuzzy sets in the fuzzy partitions. Due to not having any formal linkage between the qualitative semantics of linguistic terms and their associated fuzzy sets based semantics, any manipulation on the fuzzy sets based computational semantics is just the manipulation on

the separate mathematical objects leading to not preserve the inherent qualitative term semantics designed by human experts and effect the interpretability of FRBCs.

Hedge Algebras (HAs) [7–9] introduced by Cat Ho and Wechler [7] have rigorous efficient applications in a lot of different fields such as fuzzy control [10], data mining [11–14], image processing [15], time tabling [16], etc. HAs provide a mathematical formalism to link fuzzy sets based computational semantics of linguistic terms with their inherent qualitative semantics, in which the semantics of linguistic terms is interpreted as the order-based semantics. This formal formalism allows the fuzzy sets based computational semantics to be generated from the inherent qualitative semantics of their associated linguistic terms. Based on this basis, the first time a formalism for genetically designing linguistic terms integrated with their fuzzy sets based computational semantics in the form

of triangular membership functions for FRBCs is developed [11]. More specifically, when having the specific semantic parameter values of HAs associated with the attributes, the values of fuzziness intervals and Semantically Quantifying Mapping (SQM) are specified and all fuzzy set based computational semantics are automatically designed from SQM values by a procedure. When integrated with an optimization algorithm, this hybrid formalism allows to develop an efficient method of FRBC design. This FRBC design method comprises two phases. In the first phase, the semantic parameter values of HAs associated with data attributes are optimized by an optimization algorithm, as a result, the linguistic terms are genetically designed and a set of the optimal semantic parameter values is received. In the second phase, with the optimal semantic parameter values obtained from the first phase as an input, an optimal fuzzy classification rule set for FRBCs is genetically extracted from data based on interpretability–accuracy tradeoff. As the formalism set forth above, we can state that with Hedge Algebras methodology, the term semantics used in the fuzzy rule base representation are conservable and the semantics based measure is partially satisfied.

With ordinary Hedge Algebras [7–9], the semantic core of the linguistic term is just a value point which is SQM value of the term. In fact, each sub-value-domain of an attribute of a dataset commonly contains a value interval which is the most compatible with the qualitative semantics of linguistic term assigned to that sub-value-domain. Therefore, the representation of the semantic core of linguistic terms in the form of intervals is an indispensable requirement. In response to this requirement, ordinary Hedge Algebras is enlarged to represent the semantic core of linguistic terms in the form of intervals, so called Enlarged Hedge Algebras (EHAs) and EHAs is applied to generate trapezoidal fuzzy sets based computational semantics for FRBCs which is proved more efficient than triangular fuzzy sets based computational semantics [12].

As set forth above, the existing FRBC design methods based on Hedge Algebras approaches include two phases with each optimization process is applied separately in each phase. In the first phase, the fitness function value of an optimization process is the classification accuracy on the training set in the case of single objective and the result of the tradeoff between the classification accuracy on the training set and model complexity in the case of multiple objectives. After the first phase, the optimal semantic parameter values are received for the inputs of

the second phase. Another optimization process is applied to select the optimal fuzzy rule-based systems for FRBC in the second phase. However, the analyses of the optimization processes have shown that the best semantic parameter values in accordance with the best classification accuracy on the training set in the first phase may not give the best classification performance in the second phase, i.e., the semantic parameter values in accordance with the lower classification accuracy on the training set can make the fuzzy rule selection process give better classification performance. This paper presents a proposed co-optimization PSO algorithm for optimizing semantic parameter values and fuzzy classification rule selection concurrently. The results of the experiments executed over 23 real-world datasets have shown that Enlarged Hedge Algebras based classifier with the proposed co-optimization PSO algorithm outperforms the existing Enlarged Hedge Algebras based classifiers with two phase optimization process and the existing fuzzy set theory based classifiers.

The rest of this paper includes following sections: Section 2 presents the Enlarged Hedge Algebras based classifier design method. Section 3 presents the basic and multiple objective Particle Swarm Optimization (PSO) algorithms. Section 4 presents the proposed co-optimization PSO for solving the Enlarged Hedge Algebras based classifier design problem. The experimental results and discussion are presented in Section 5. The conclusion is marked in Section 6.

## 2 Enlarged Hedge Algebras (EHAs) based classifier design method

### 2.1 Enlarged Hedge Algebras (EHAs) for modeling semantic core of linguistic terms

Given a linguistic variable  $X$  and its linguistic value domain is  $\text{Dom}(X)$ . A Hedge Algebras  $\mathcal{AX}$  of  $X$  is a structure  $\mathcal{AX} = (X, G, C, H, \leq)$ , where  $X$  is the set of linguistic terms of  $X$ ;  $G = \{c^-, c^+\}$  is a set of two generators, where  $c^-$  and  $c^+$  are the negative and positive generator term, respectively, and  $c^- \leq c^+$ ;  $C = \{0, W, 1\}$  is a term constant set, where 0,  $W$  and 1 is the least, neutral and greatest term, respectively, satisfying the semantic order relation  $0 \leq c^- \leq W \leq c^+ \leq 1$ ;  $H$  is a set of linguistic hedges and  $H = H^- \cup H^+$ , where  $H^- = \{h_{-q}, \dots, h_{-1}\}$  and  $H^+ = \{h_1, \dots, h_p\}$  are the set of negative and positive linguistic hedges, respectively, satisfying the order relation  $h_{-q} \leq \dots \leq h_{-1} \leq h_1 \leq \dots \leq h_p$ ;  $\leq$  is the semantic order relation which is induced by inherent qualitative semantics of terms of  $X$ .

A new linguistic term is induced by acting a linguistic hedge on a non-constant linguistic term. Each linguistic term is represented as a string, i.e., either  $x = h_n \dots h_1 c$  or  $x = c$ , where  $h_i \in H, i = 1, \dots, n$  and  $c \in \{c^-, c^+\} \cup C$ . All linguistic terms induced from  $x$  by using linguistic hedges in  $H$  are abbreviated as  $H(x)$ . In case all hedges in  $H$  are linear ordered and induced all linear ordered linguistic terms,  $\mathcal{AX}$  is linear Hedge Algebras. We just examine linear Hedge Algebras, so it is just called Hedge Algebras or HAs for short.

Each linguistic hedge has its tendency to increase or decrease semantics of the other hedges. A hedge  $k$  is positive with respect to  $h$  and has  $\text{Sign}(k, h) = +1$  if  $k$  makes the semantic of  $h$  increased. Whereas, a hedge  $k$  is negative with respect to  $h$  and has  $\text{Sign}(k, h) = -1$  if  $k$  makes the semantic of  $h$  decreased. The positivity and negativity of the hedges do not have any dependence on linguistic terms on which they act. So, the sign of a linguistic term  $x = h_n h_{n-1} \dots h_1 h_2 c$  is computed as:

$$\text{Sign}(x) = \text{Sign}(h_n, h_{n-1}) \times \dots \times \text{Sign}(h_2, h_1) \times \text{Sign}(h_1) \times \text{Sign}(c).$$

The sign of term has meaning: if  $\text{Sign}(hx) = +1$  then  $x \leq hx$  and if  $\text{Sign}(hx) = -1$  then  $hx \leq x$ .

In [12], Enlarged Hedge Algebras (EHAs) is extended from ordinary linear Hedge Algebras [7–9] by adding an artificial hedge  $h_0$  for modeling semantic core of linguistic terms. A new term  $h_0 x$  is induced by acting  $h_0$  on  $x \in X$  and has its property: after  $h_0$  acts on  $x$ ,  $h_0 x$  becomes term constant, i.e.,  $\sigma h_0 x = h_0 x$ , where  $\sigma \in H^{en} = H \cup h_0$ .

A structure  $\mathcal{AX}^{en} = (X^{en}, G, C, H^{en}, \leq)$ , where  $H^{en} = H \cup h_0$ , is called Enlarged Hedge Algebras (EHAs) if it satisfies the following additional axioms:

- $h_0 x \notin H(G) = \{\sigma c | c \in G\}$  and  $h h_0 x = h_0 x$  is always a fixed point.
- $h_p x \geq x \Rightarrow h_{-q} \leq \dots \leq h_{-1} x \leq h_0 x \leq h_1 x \leq \dots \leq h_p x$   
 $h_p x \leq x \Rightarrow h_p x \leq \dots \leq h_1 x \leq h_0 x \leq h_{-1} x \leq \dots \leq h_{-q} x$ .

The fuzziness measures of term constants can be greater than 0. So, some axioms should be extended to adapt to new structure of  $\mathcal{AX}^{en}$  and fuzziness measure of  $h_0$ .

**Definition 1** [12]. A function  $fm: X^{en} \rightarrow [0, 1]$  is called the fuzziness measure of  $\mathcal{AX}^{en}$  if it satisfies properties as follows:

- $fm(0) + fm(c^-) + fm(W) + fm(c^+) + fm(1) = 1$ ;
- $\sum_{x \in H^{en}} fm(hx) = fm(x), \forall x \in H(G)$ ;

- $\forall x, y \in H(G), \forall h \in H^{en}$ , the proportion  $\frac{fm(hx)}{fm(x)} = \frac{fm(hy)}{fm(y)}$  which does not have dependence on any linguistic term of  $X^{en}$  is called the fuzziness measure of the hedge  $h$ , denoted by  $\mu(h)$ .

From Definition 1, the fuzziness measure of linguistic term  $x = h_n \dots h_1 c$  can be calculated recursively as  $fm(x) = \mu(h_n) \times \dots \times \mu(h_1) \times fm(c)$ , where  $\sum_{h \in H^{en}} \mu(h) = 1$  and  $c \in \{c^-, c^+\}$ .

**Proposition 1** [12]. A fuzziness measure of a linguistic term of EHAs  $\mathcal{AX}^{en}$  satisfying the following properties:

- $\sum_{x \in X(k)} fm(x) = 1, k > 0$ . In case  $k = 1$ , we have  $fm(0) + fm(c^-) + fm(W) + fm(c^+) + fm(1) = 1$ ;
- $\sum_{h \in H^{en}} \mu(h) = 1$ ;
- $fm(hx) = \mu(h) fm(x)$ , for  $\forall h \in H^{en}, \forall x \in H(\{c^-, c^+\})$  and  $hx \neq x$ ;
- $fm(hx) = \mu(h_n) \dots \mu(h_1) fm(c)$ , where  $x = h_n \dots h_1 c, c \in \{c^-, c^+\}$ , is string presentation of  $x \in X^{en}$ .

**Definition 2** [12]. Given fuzziness measure  $fm: X^{en} \rightarrow [0, 1]$  of EHAs  $\mathcal{AX}^{en}$  of a linguistic variable  $X$  and each term  $x \in X^{en}$  is mapped to an interval  $\mathfrak{I}(x) \subseteq [0, 1]$ . These intervals are called fuzziness intervals of linguistic terms of  $X$  provided that:

- $|\mathfrak{I}(x)| = fm(x), \forall x \in X^{en}$ , where  $|\mathfrak{I}(x)|$  denotes the length of  $\mathfrak{I}(x)$ ;
- The set  $\{\mathfrak{I}(hx) | x \in X^{en}\}$  is a partition of  $|\mathfrak{I}(x)|$  and their order relation is the same order relation of their associated linguistic terms.

$PI([0, 1])$  denotes all sub-intervals of  $[0, 1]$ .

**Definition 3** [12]. Given  $\mathcal{AX}^{en}$  is a linear Enlarged Hedge Algebras, a mapping  $f: X^{en} \rightarrow PI([0, 1])$  is called interval Semantically Quantifying Mapping of  $\mathcal{AX}^{en}$  provided that:

- $f$  preserves the order relation on  $X^{en}$ , i.e., if  $x \leq y$  then  $f(x) \leq f(y)$ , for  $\forall x, y \in X^{en}$ ;
- $f(X^{en})$  is dense in  $[0, 1]$ .

**Theorem 1** [12].  $\mathfrak{I}$  is a set of all fuzziness intervals of  $\mathcal{AX}^{en}$ . A mapping  $f: X^{en} \rightarrow \mathfrak{I} \subseteq PI([0, 1])$  defined as follows is interval Semantically Quantifying Mapping: For  $\forall x \in X^{en}, f(x) = \mathfrak{I}_{|x|+1}(h_0 x) \subseteq PI([0, 1])$  with noting that if  $x = h_0 z$  then  $f(x) = \mathfrak{I}_{|x|+1}(h_0 x) = \mathfrak{I}_{|x|}(h_0 z)$ , where  $|x|$  denotes the length of  $x$ .

### 2.2 Fuzzy Rule-Based Classifier (FRBC) design based on Enlarged Hedge Algebras (EHAs)

A Fuzzy Rule-Based Classifier design problem  $\mathcal{P}$  is defined as: a dataset  $P = \{(\mathbf{d}_p, C_p) | \mathbf{d}_p \in D, C_p \in \mathbf{C}, p = 1, \dots, m\}$  of  $m$  patterns, where  $\mathbf{d}_p = [d_{p,1}, d_{p,2}, \dots, d_{p,n}]$  is the row  $p^{\text{th}}$ ;  $n$  is the number of attributes of  $P$ ;  $C_l$  is a class label,  $l = 1, \dots, M$ .

The weighted fuzzy rules of FRBCs exploited in this paper have the form as followings [4, 5]:

Rule  $R_q$ : **If**  $\mathcal{X}_1$  is  $A_{q,1}$  **and** ... **and**  $\mathcal{X}_n$  is  $A_{q,n}$  **then**  $C_q$  (1)  
 with  $CF_q$ , for  $q = 1, \dots, N$

where  $\mathcal{X}_j$  is a linguistic variable associated with an attribute of  $P$ ,  $j = 1, \dots, n$ ;  $A_{q,j}$  is a linguistic term;  $C_q$  is a class label;  $CF_q$  is the rule weight of  $R_q$ . The short form of  $R_q$  is in Eq. (2):

$$A_q \Rightarrow C_q \text{ with } CF_q, \text{ for } q = 1, \dots, N. \quad (2)$$

The problem  $\mathcal{P}$  is solved by extracting from  $P$  a compact fuzzy rule set  $\mathcal{S}$  in Eq. (1) which has a good tradeoff between classification accuracy and model complexity. The classifier design method based on Enlarged Hedge Algebras methodology is summarized as follows [11, 12].

Because the interval semantics quantifying mapping  $f(x_{j,i}) \subseteq \mathfrak{S}_{k_j}(x_{j,i})$  is the semantic core presentation of  $x_{j,i}$ , so  $f(x_{j,i})$  is compatible with the core (small base) of trapezoidal fuzzy set based computational semantics of  $x_{j,i}$ . The multi-granularity structure of fuzzy partitions proposed in [10] is depicted in Fig. 1.

Each EHAs  $\mathcal{A}\mathcal{X}_j^{en}$  associated with an attribute  $j$  of designated dataset induces entire linguistic terms  $X_{j,(k_j)}$  with the length from 1 to  $k_j$ , and have their own qualitative semantic order relation. When given the values of

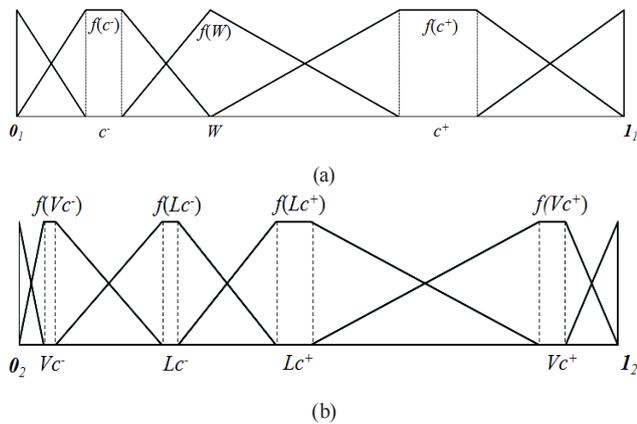


Fig. 1 Multi-granularity structure of fuzzy partition. Fuzzy partition just has linguistic terms (a) with the length 1, (b) with the length 2

$fm(c_j^-), fm(W_j), fm(0_j), fm(1_j), \mu(h_{j,i}), \mu(h_{j,0})$  which are the fuzziness measures of  $c_j^-, W_j, 0_j, 1_j, h_{j,i}, h_{j,0}$ , respectively, and  $k_j$  specifies the maximal length of linguistic terms, the fuzziness intervals  $\mathfrak{S}_{k_j}(x_{j,i})$  and the interval semantic quantifying mapping  $f(x_{j,i})$  of  $x_{j,i} \in X_{j,k_j}(0 < k \leq k_j)$  are computed. The fuzziness intervals  $\mathfrak{S}_{k_j}(x_{j,i})$  form a fuzzy partition at level  $k_j$  on the value domain of attribute  $j$ . There is only one fuzziness interval  $\mathfrak{S}_{k_j}(x_{j,i(i)})$  in  $\mathfrak{S}_{k_j}(x_{j,i})$  containing  $j^{\text{th}}$ -component  $d_{p,j}$  of  $\mathbf{d}_p$  pattern. All fuzziness intervals at level  $k_j$  containing  $d_{p,j}$  ( $0 < j \leq n$ ) specify a hypercube from which fuzzy rules can only be generated. Fuzzy rules which have the length  $n$  are called basic fuzzy rules and have the form as follows:

**If**  $\mathcal{X}_1$  is  $x_{1,i(1)}$  **and** ... **and**  $\mathcal{X}_n$  is  $x_{n,i(n)}$  **then**  $C_p(R_b)$ .

The secondary rules which have the length  $L \leq n$  are generated by eliminated  $n - L$  attributes from basic rules and have the form as follows:

**If**  $\mathcal{X}_{j_1}$  is  $x_{j_1,i(j_1)}$  **and** ... **and**  $\mathcal{X}_{j_l}$  is  $x_{j_l,i(j_l)}$  **then**  $C_q(R_{snd})$

where  $1 \leq j_1 \leq \dots \leq j_l \leq n$ . Class label  $C_q$  of  $R_q$  is specified by the confident  $c(A_q \Rightarrow C_h)$  of  $R_q$  [4, 5]:

$$C_q = \arg \max \{c(A_q \Rightarrow C_h) | h = 1, \dots, M\}. \quad (3)$$

The rule confident is calculated by Eq. (4):

$$c(A_q \Rightarrow C_h) = \frac{\sum_{\mathbf{d}_p \in C_h} \mu_{A_q}(\mathbf{d}_p)}{\sum_{p=1}^m \mu_{A_q}(\mathbf{d}_p)} \quad (4)$$

where  $\mu_{A_q}(\mathbf{d}_p)$  is the burning of data pattern  $\mathbf{d}_p$  with the antecedent of  $R_q$  and calculated by Eq. (5):

$$\mu_{A_q}(\mathbf{d}_p) = \prod_{j=1}^n \mu_{q,j}(\mathbf{d}_{p,j}). \quad (5)$$

A set of  $S_0$  rules which is so-called the initial rule set is selected by a screening criterion. The commonly used screening criterion is  $c \times s$ . However, the confident  $c$  and the support  $s$  are used in some cases. The number rules in initial rule set is  $NR_0 = NB_0 \times M$ , where  $M$  and  $NB_0$  are the number of classes and the number of rules in each class, respectively. The confident is calculated by Eq. (4), the support is calculated by Eq. (6) [4]:

$$s(A_q \Rightarrow C_h) = \frac{\sum_{\mathbf{d}_p \in C_h} \mu_{A_q}(\mathbf{d}_p)}{m}. \quad (6)$$

The rule weight used to improve the classification accuracy is calculated in this research by Eq. (7) [4]:

$$CF_q = c(A_q \Rightarrow C_q) - c_{q,2nd} \quad (7)$$

where  $c_{q,2nd}$  is the maximum confident of the fuzzy rules which have the same antecedent  $A_q$  and have different class label  $C_q$ :

$$c_{q,2nd} = \max \{c(A_q \Rightarrow \text{Class } h) | h = 1, \dots, M; h \neq C_q\}. \quad (8)$$

The process described above is called the initial rule set generation procedure **IFRG**( $\pi, P, NR_0, L$ ) [11, 12], where  $\pi$  is the set of input values of the semantic parameters and  $L$  is used to limit the maximum length of rule antecedents.

In the first phase of the fuzzy rule-based classifier design method based on HAs, the procedure **IFRG** is used to generate an initial rule set in each individual of the applied optimization algorithm in order to receive a set of semantic parameter values in accordance with the highest classification accuracy on the training set which is so-called the optimal semantic parameter values. In the second phase, the procedure **IFRG** is just used once to generate an initial rule set for the process of the optimal fuzzy rule selection [11, 12, 17].

### 3 Particle Swarm Optimization (PSO)

#### 3.1 Standard Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) proposed by Kennedy and Eberhart in 1995 [18, 19] has been used as an efficient optimization algorithm to a lot of real world problems. Individuals and population are called particles and swarm, respectively. Each particle in the swarm moves in a search space with a velocity computed by its own and its group previous best solutions.

Assume that there is a swarm  $S = \{x_1, x_2, \dots, x_N\}$ , where  $N$  is the number of particles,  $X_i^t$  is the position of particle  $i$  in the search space at generation  $t$  and updated using Eq. (9):

$$X_i^{t+1} = X_i^t + V_i^{t+1}, \quad (9)$$

where  $V_i^{t+1}$  is the velocity of particle  $i$  at generation  $t + 1$  updated in Eq. (10):

$$V_i^{t+1} = \omega V_i^t + c_1 r_1 (P_i^t - X_i^t) + c_2 r_2 (P_g^t - X_i^t), \quad (10)$$

where  $P_i^t$  and  $P_g^t$  are the best local and global solutions found up to the generation  $t$ , respectively. Two uniform random numbers  $r_1$  and  $r_2$  are distributed in the normalized interval  $[0, 1]$ . The  $c_1$  is self-cognitive factor and  $c_2$  is social cognitive factor. The  $\omega$  is the inertia weight. The formal algorithm of the standard PSO is abbreviated in Algorithm 1.

#### 3.2 Multi-objective PSO with fitness sharing

Basic PSO just supports single-objective problems (SOO), so many studies have been carried out to improve it to

---

#### Algorithm 1 Standard PSO algorithm

---

- Step 1: Initialize the cycle  $t$ , generate swarm  $S$  randomly within the search space.
  - Step 2: Calculate the objective value  $f(x_i)$  for all particles.
  - Step 3: Update the personal best  $P_i^t$  for all particle.
  - Step 4: Update the global best  $P_g^t$ .
  - Step 5: Calculate the particle velocities by Eq. (10).
  - Step 6: Move particles to their new positions by Eq. (9).
  - Step 7: Increase the cycle variable.
  - Step 8: Go to step 2 and repeat until convergence or the max value of  $t$  reached.
- 

support multi-objective problems (MOO). One of them is the multi-objective PSO with fitness sharing proposed in [20].

Fitness sharing  $fshare_i$  for a particle  $i$  is defined by Eq. (11):

$$fshare_i = \frac{f_i}{\sum_{j=0}^n sharing_j^i} \quad (11)$$

where  $n$  is the number of particles in swarm and  $sharing_j^i$  is calculated by Eq. (12):

$$sharing_j^i = \begin{cases} 1 - (d_i^j / \sigma_{share})^2 & \text{If } d_i^j < \sigma_{share} \\ 0 & \text{Otherwise} \end{cases} \quad (12)$$

where  $\sigma_{share}$  is the distance which particles should remain,  $d_i^j$  means the distance between particle  $i$  and  $j$ .

$$d_i^j = \sqrt{(\text{particle}_i - \text{particle}_j)^2} \quad (13)$$

The *Pareto dominance* concept is used to maintain a set of best solutions so far. The concepts of *non-dominated set* and *Pareto dominance* can be found in [20].

The brief explanation of multi-objective PSO algorithm with fitness sharing is described in Algorithm 2 (the detail is in [20]).

### 4 Co-optimization PSO for Hedge Algebras based Classifier Design Methods (HACDMs)

As mentioned above, the existing Hedge Algebras based Classifier Design Methods (HACDMs) [11, 12] comprise two phases. The first phase is merely for optimizing semantic parameter values. The second phase is merely for selecting the optimal fuzzy rule set for FRBCs. The disadvantage of the two phase design method is that the optimal semantic parameter values received from the first phase may not give the best classification performance in the second phase. To tackle this disadvantage, Section 4 presents a proposed optimization algorithm for co-optimizing semantic parameter values and fuzzy classification rule system. More specifically, after each optimization cycle

---

**Algorithm 2** Multi-objective PSO algorithm
 

---

Initialize all global variables ( $X_i, pbest_i, gbest_i, fshare_i$ ).  
 Evaluate the objective values of all particles. Fitness sharing value of each particle is calculated as:

$$fshare_i = \frac{x}{nCount_i},$$

Step 1: where  $x = 10, nCount_i$  value is calculated as:

$$nCount_i = \sum_{j=0}^n sharing_i^j,$$

where  $n$  is the number of non-dominated particles stored in the external archive and  $sharing_i^j$  value is calculated by Eq. (12).

Step 2: Calculate new particle velocities by Eq. (10).

Step 3: Calculate new particle positions by Eq. (9).

Step 4: Evaluate fitness values of all objectives of particles.

Step 5: Update external archive by the concepts of dominance and fitness sharing (see [20]).

Step 6: Update the memory of each particle based on the dominance criteria (see [20]).

Step 7: The algorithm terminates when the termination condition is reached. Otherwise, go to step 2.

---

(generation)  $t$  of the semantic parameter value optimization process, an optimal fuzzy classification rule system selection process is executed with the best semantic parameter values according to the best classification accuracy on the training set among individuals of current cycle  $t$  as the inputs. After each fuzzy classification rule system selection process, the best fuzzy rule sets according to the best classification performance (the best tradeoff between classification accuracy and model complexity) on training set are compared to the ones in the archive to ensure that only the best ones so far are archived.

In our implementation, a single objective PSO is applied in the semantic parameter value optimization cycles with the fitness function is in Eq. (14):

$$accu(Cla(\mathcal{S}_0(\pi))) \rightarrow \text{Max} \quad (14)$$

where  $Cla(\mathcal{S}_0(\pi))$  is a classifier which uses the procedure **IFRG**( $\pi, P, NR_0, L$ ) to generate the initial rule set  $\mathcal{S}_0$  and  $accu$  denotes the classification accuracy on training set. During the learning process, the semantic parameter values should satisfy the constraints:  $a_j \leq fm(c_j^-) \leq a'_j, b_j \leq fm(W_j) \leq b'_j, fm(0_j) + fm(c_j^-) + fm(W_j) + fm(c_j^+) + fm(1_j) = 1,$

$$e_j \leq \mu(h_{j,i}) \leq e'_j, \quad \sum_{h_{j,i} \in H_j} \mu(h_{j,i}) = 1, k_j \leq L, j = 1, \dots, n,$$

where  $n$  is the number of attributes of the designated dataset,  $fm$  and  $\mu$  defined in Definition 1 denote the fuzziness measures of linguistic terms and linguistic hedges, respectively.

The multi-objective PSO algorithm set forth above is applied in the optimal fuzzy rule selection cycles to select a subset of rule  $\mathcal{S}$  from  $\mathcal{S}_0$  satisfying the objectives defined by Eq. (15):

$$accu(Cla(\mathcal{S})) \rightarrow \text{Max}, NR(\mathcal{S}) \rightarrow \text{Min}, avgrl(\mathcal{S}) \rightarrow \text{Min}, \text{satisfying constraints } \mathcal{S} \subset \mathcal{S}_0, NR(\mathcal{S}) \leq N_{\max} \quad (15)$$

where  $NR(\mathcal{S})$  and  $avgrl(\mathcal{S})$  are the number of fuzzy rules in  $\mathcal{S}$  and the average rule length, respectively, and  $N_{\max}$  is a pre-defined positive integer used to limit the number of fuzzy rules in  $\mathcal{S}$  during training process. The real encoding of particle is used where each particle corresponds to a solution represented as a string of real number  $r_i = (p_1, \dots, p_{N_{\max}}, p_j \in [0,1])$ . Each fuzzy rule  $R_j$  of  $\mathcal{S}$  is selected from  $\mathcal{S}_0$  by zero based index calculated by Eq. (16):

$$\mathcal{S} = \{R_i \in \mathcal{S}_0 \mid i = \lfloor p_j \times |\mathcal{S}_0| \rfloor, i \geq 0\} \quad (16)$$

where  $\lfloor \cdot \rfloor$  denotes integer part of a real number.

The general diagram of our proposed co-optimization PSO algorithm is depicted in Fig. 2. The algorithm in detail is described in Algorithm 3.

The output of the co-optimization PSO algorithm is a set of the optimal solutions, from which the best one is chosen. The chosen solution corresponds to the fuzzy rule set which has the best classification accuracy on training set and low complexity measured by the product of the average rule length and the number of fuzzy rules.

**Remark:** The single PSO algorithm which makes the outer iterations can be enhanced to reduce its running time and reduce the total running time of Algorithm 3. Because the fitness function of single PSO may not be better after several iterations (generations), the semantic parameter values are also kept unchanged. Therefore, it had better limit to call multi-objective PSO in case the fitness function of single PSO is not enhanced after some generations (after three generations in our implementation).

## 5 Experimental results and discussion

Section 5 presents the analyses of experimental results of our proposed classifier which the co-optimization PSO algorithm is applied to concurrently optimize semantic parameter values and fuzzy rule systems and show that it is better than the existing Hedge Algebras based design methods and other design methods based on fuzzy set theory.

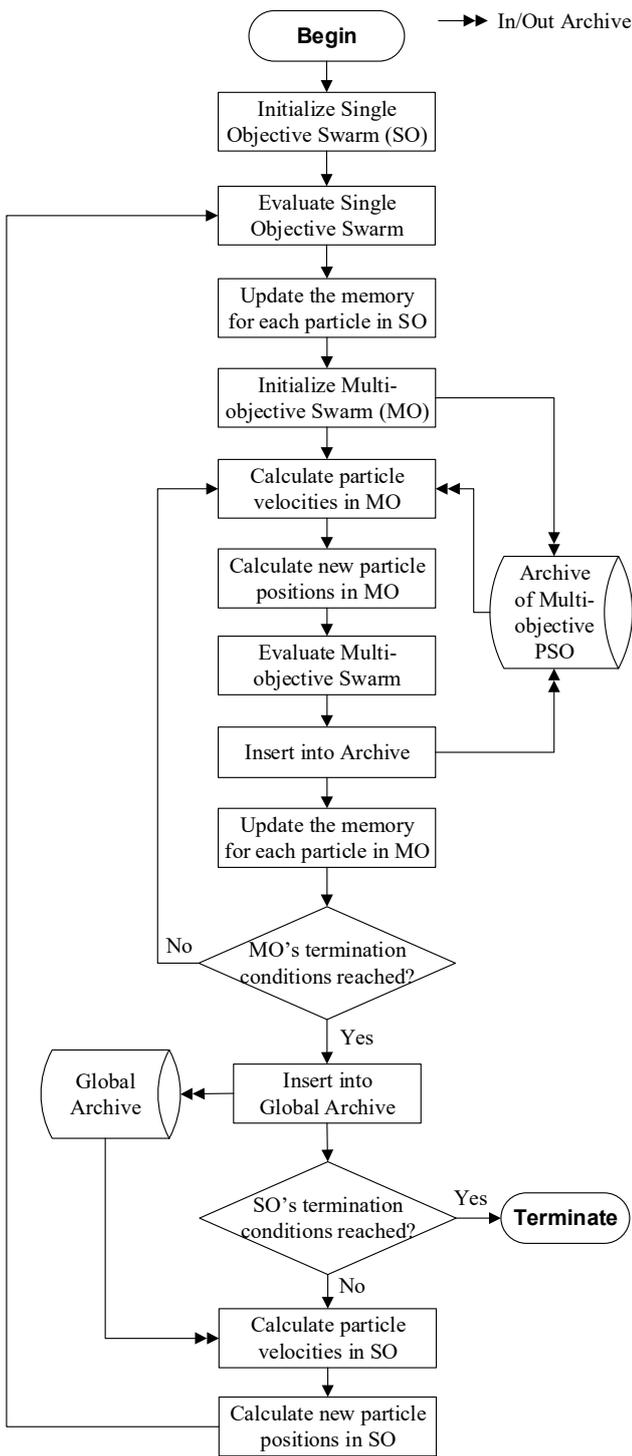


Fig. 2 General diagram of proposed co-optimization PSO

### 5.1 Experiment setup

Our experiments have been implemented using C# running on Microsoft Windows 10. The experimented real-world datasets shown in Table 1 come from KEEL-dataset repository at address [21]. The ten-fold cross validation method is applied to every validated dataset and the partitioned folds can be also found at [21]. Three ten-folds cross validations

### Algorithm 3 Co-optimization PSO algorithm for FRBCs

	The dataset $P = \{(d_p, C_p)   p = 1, \dots, m\}$ ;
	Parameters: $NR_{SO}, NR_{MO}, N_{SO}, N_{MO}, G_{Smax}, G_{Mmax}, L_{SO}, L_{MO}$ . $//N_{SO}$ and $N_{MO}$ are the swarm sizes of single and multiple objective PSO, respectively. $//G_{Smax}$ and $G_{Mmax}$ are the number of generations of single and multiple objective PSO, respectively. $//L_{SO}$ and $L_{MO}$ specify the max length of linguistic terms in single and multiple objective PSO, respectively.
Input:	
Output:	the optimal fuzzy rule-based systems for FRBCs.
Step 1:	Randomly initialize a single objective swarm $PSO_i = \{\pi_{t,i}   i = 1, \dots, N_{SO}\}, t = 0$ . Evaluate single objective swarm which includes generating the fuzzy rule set $S_0(\pi_{t,i})$ from $\pi_{t,i}$ by applying the initial fuzzy rule generation procedure $IFRG(\pi_{t,i}, P, NR_{SO}, L_{SO})$ ; Evaluating the objective value for all particles by Eq. (14).
Step 2:	
Step 3:	Update the memory for all particles and get the best semantic parameter values $\pi_r$ according to the best classification accuracy on the training set.
Step 4:	Jump to multi-objective PSO by randomly initializing a multi-objective swarm with the size $N_{MO}$ . So, all global variables are initialized. The fitness sharing value for each particle is calculated. Generate initial rule set $S_0(\pi_r)$ from $\pi_r$ by applying $IFRG(\pi_r, P, NR_{MO}, L_{MO})$ .
Step 5:	Calculate new particle velocities by Eq. (10).
Step 6:	Calculate new particle positions by Eq. (9).
Step 7:	Evaluate multi-objective swarm which includes calculating all objective function values ( $accu(Cla(S_i)), NR(S_i), avgrl(S_i), i = 1, \dots, N_{MO}$ ) from subset $S$ selected from $S_0$ based on the position of each particle.
Step 8:	Get the best fuzzy rule set according to the best tradeoff between classification accuracy on the training set and the complexity of fuzzy rule bases and insert into the local archive $LoArc$ by fitness sharing and dominance concepts.
Step 9:	Update the local memory for all particles by dominance criteria.
Step 10:	If the number of iterations $G_{Mmax}$ of multi-objective PSO is reached, go to next step. Otherwise, increase the iteration variable and go to Step 5.
Step 11:	Insert a set of the best fuzzy rule set from the local archive $LoArc$ of multi-PSO into the global archive $GLArc$ based on the dominance criteria.
Step 12:	Jump back to single objective PSO. If the number of iterations $G_{Smax}$ of single objective PSO is reached ( $t = G_{Mmax}$ ), the algorithm terminates. Otherwise, increase the iteration variable $t$ and go to the next step.
Step 13:	Calculate the velocity for particles.
Step 14:	Calculate new positions for particles. Go to Step 2.

are executed for each dataset and, hence, it permits to extract 30 ( $3 \times 10$  folds) fuzzy rule-based systems for FRBCs. The Wilcoxon Signed Rank Test (WSRT) [22] is used to detect the significant differences between the tested methods.

To reduce the search space during the training processes, some constraints should be imposed on the semantic parameter values as follows:

**Table 1** The datasets used in our experiments

No.	Dataset name	Short name	No. of attributes	No. of classes	No. of patterns
1	Appendicitis	App	7	2	106
2	Australian	Aus	14	2	690
3	Bands	Ban	19	2	365
4	Bupa	Bup	6	2	345
5	Cleveland	Cle	13	5	297
6	Dermatology	Der	34	6	358
7	Glass	Gla	9	6	214
8	Haberman	Hab	3	2	306
9	Hayes-roth	Hay	4	3	160
10	Heart	Hea	13	2	270
11	Hepatitis	Hep	19	2	80
12	Ionosphere	Ion	34	2	351
13	Iris	Iri	4	3	150
14	Mammogr.	Mam	5	2	830
15	Newthyroid	New	5	3	215
16	Pima	Pim	8	2	768
17	Saheart	Sah	9	2	462
18	Sonar	Son	60	2	208
19	Tae	Tae	5	3	151
20	Vehicle	Veh	18	4	846
21	Wdbc	Wdb	30	2	569
22	Wine	Win	13	3	178
23	Wisconsin	Wis	9	2	683

- The number of positive and negative hedges is 1, positive hedge is *Very (V)* and negative hedge is *Less (L)*;  $1 \leq k_j \leq 3$ ;
- $0.2 \leq \{fm(c_j^-), fm(c_j^+)\} \leq 0.7$ ;  
 $0.00001 \leq \{fm(0_j), fm(1_j)\} \leq 0.1$ ;  
 $0.0001 \leq fm(W_j) \leq 0.2$ ;
- $fm(0_j) + fm(c_j^-) + fm(W_j) + fm(c_j^+) + fm(1_j) = 1$ ;
- $0.2 \leq \{\mu(L_j), \mu(V_j)\} \leq 0.7$ ;  $0.01 \leq \mu(h_{0,j}) \leq 0.5$
- $\mu(L_j) + \mu(h_{0,j}) + \mu(V_j) = 1$ .

The parameter values of co-optimization PSO algorithm: Inertia weight is 0.4, self-cognitive factor is 0.2 and the number of particles in the swarm is 600.

- Single objective PSO: the number of cycles is 250, social cognitive factor is 0.2, the max rule length is 1, the number of rules in initial rule set is equal to the number of attributes.

- Multi-objective PSO: the number of cycles is 500, social cognitive factor is 0.1, the max rule length is 3, the number of rules in initial rule set is  $no\_attrs \times no\_labels \times 10$ , where  $no\_attrs$  is the number of attributes and  $no\_labels$  is the number of class labels.

The classification reasoning method used in all experiments is *single winner rule* [4, 5]. The screening criterion is  $c \times s$ , where  $c$  and  $s$  are the confident and the support, respectively. The rule weight is calculated by Eq. (7).

### 5.2 Results and discussion

As discussed above, with the two phase design method [10], the optimization process of the second phase does not always give out the optimal fuzzy rule-based system for FRBCs providing that the so-called optimal semantic parameter values received from the first phase are the inputs of the second phase for initial fuzzy rule set generation. This supposition is clarified with the analysis of co-optimization process data of each iteration. For the given Wine dataset, the statistical data of the run 2 with 250 outer iterations is shown in Table 2, where  $\#R, \#R \times C, P_{tr}$  and  $P_{te}$  denote the average values of the number of fuzzy rules, model complexity (the product of the average values of the number of fuzzy rules and the number of rule conditions), the classification accuracy on the training set and the classification accuracy on the testing set, respectively. It can be seen that there are a half of ten folders which the semantic parameter values received from the outer iteration according to the less classification accuracy on the training set, so-called the optimal training accuracy, give the optimal fuzzy rule set in the inner iteration. It proves that the semantic parameter values according to the best classification accuracy on the training set, so-called the

**Table 2** The statistical data of the run 2 of Wine dataset

Folder	The optimal iteration	The optimal training accuracy	The best training accuracy
1	46	99.44 %	99.44 %
2	23	98.88 %	99.44 %
3	38	98.88 %	98.88 %
4	4	98.88 %	98.88 %
5	179	98.88 %	98.88 %
6	196	99.44 %	99.44 %
7	5	98.31 %	98.88 %
8	57	99.44 %	100 %
9	14	97.75 %	98.31 %
10	11	97.75 %	98.31 %

best training accuracy, does not always give the optimal fuzzy rule set for FRBCs. For example, intuitively seen in Table 2 that folder 2 reaches the optimal training accuracy 98.88 % at the iteration 23, so-called the optimal iteration, less than the best training accuracy 99.44 %.

For more convenience, the classifier with two phase design method is denoted by HATF and the classifier with co-optimization design method is denoted by HACO. The experimental results and comparison between two classifiers on the testing sets and the model complexity are shown in Table 3. Intuitively seen that HACO has better classification accuracies on 20 of 23 experimented datasets. Considering on the mean values, HACO has higher mean value of classification accuracy and lower model complexity than the one of HATF (82.95 % and 112.73 in comparison with 82.67 % and 114.78, respectively).

To ensure the significant difference between two experimental results, Wilcoxon Signed Rank Test [22] at level is used test the equivalent hypothesis. The test of classification accuracies in Table 4 shows that HACO is better than HATF on classification accuracy because the

$p$ -value = 0.0011184 is less than  $\alpha = 0.05$ , so the equivalent hypothesis is rejected and the mean value of classification accuracy of HACO is greater than the one of HATF. The test of model complexity in Table 5 shows that HACO and HATF have the same model complexity because the  $p$ -value is greater than  $\alpha = 0.05$ , so the equivalent hypothesis is not rejected. Based on the test results of both classification accuracy and model complexity, we can state that HACO outperforms HATF.

To show that our proposed classifier is better than the existing classifiers designed based on the fuzzy set theory such as Alcalá et al. [1] so-called Product-1-ALL TUN, Antonelli et al. [2] so-called PAES-RCS as well as compared with a non-evolutionary classification algorithm so-called FURIA, the experimental results of them are compared with one another.

In [1], Alcalá et al. proposed several techniques to select the single granularity from the predesigned multi-granularities for genetically extracting fuzzy rules for FRBCs. The best technique which has the membership function parameter value tuning concurrently with

**Table 3** The experimental results and comparison between HACO and HATF classifiers

No.	Dataset name	HACO				HATF				$\neq R \times C$	$\neq P_{te}$
		#R	#R $\times$ C	$P_{tr}$	$P_{te}$	#R	#R $\times$ C	$P_{tr}$	$P_{te}$		
1	Appendicitis	3.93	19.65	91.79	89.09	3.67	16.77	92.38	88.15	2.88	0.94
2	Australian	5.67	51.99	88.27	87.20	5.00	46.50	88.56	87.15	5.49	0.05
3	Bands	6.00	56.40	76.39	73.00	6.00	58.20	78.19	73.46	-1.80	-0.46
4	Bupa	10.33	226.95	76.22	73.22	8.97	181.19	79.78	72.38	45.76	0.84
5	Cleveland	14.70	465.55	70.34	62.12	14.57	468.13	66.64	62.39	-2.59	-0.27
6	Dermatology	11.93	229.41	97.36	94.96	10.43	182.84	96.37	94.40	46.58	0.56
7	Glass	13.50	357.75	79.15	73.07	14.23	474.29	78.78	72.24	-116.54	0.83
8	Haberman	3.00	9.81	76.86	77.50	3.00	10.80	77.60	77.40	-0.99	0.10
9	Hayes-roth	10.17	111.87	90.46	84.79	9.80	114.66	89.40	84.17	-2.79	0.62
10	Heart	7.63	97.44	87.92	84.94	8.37	123.29	89.19	84.57	-25.86	0.37
11	Hepatitis	3.87	20.63	92.17	89.15	3.70	25.53	93.68	89.28	-4.90	-0.13
12	Ionosphere	8.90	102.35	94.81	91.64	8.63	88.03	94.69	91.56	14.32	0.08
13	Iris	4.53	22.97	98.17	98.00	5.30	30.37	98.25	97.33	-7.40	0.67
14	Mammogr.	7.17	75.50	85.88	84.25	7.10	73.84	85.49	84.20	1.66	0.05
15	Newthyroid	5.87	46.78	97.95	95.70	5.33	39.82	96.76	95.67	6.97	0.03
16	Pima	6.93	76.23	78.41	77.22	5.97	56.12	78.69	77.01	20.11	0.21
17	Saheart	6.70	75.04	76.28	70.26	5.63	59.28	75.51	70.05	15.76	0.21
18	Sonar	5.93	46.43	87.78	78.94	5.87	49.31	87.59	78.61	-2.88	0.33
19	Tae	9.93	157.89	71.08	61.67	10.90	210.70	68.97	61.00	-52.81	0.67
20	Vehicle	11.00	178.20	70.67	68.32	11.23	195.07	70.74	68.20	-16.87	0.12
21	Wdbc	4.93	36.83	97.53	96.81	4.00	25.04	97.08	96.78	11.79	0.03
22	Wine	5.87	44.20	99.77	99.05	5.77	40.39	99.60	98.49	3.81	0.56
23	Wisconsin	8.47	83.01	98.01	96.99	7.87	69.81	97.78	96.95	13.20	0.04
Mean			112.73	86.23	82.95		114.78	86.16	82.67		

**Table 4** The equivalent hypothesis test result on classification accuracy of HACO and HATF

VS	$R^+$	$R^-$	E. $P$ -value	Hypothesis
HACO vs HATF	240	36	0.0011184	Rejected

**Table 5** The equivalent hypothesis test result on model complexity of HACO and HATF

VS	$R^+$	$R^-$	E. $P$ -value	Hypothesis
HACO vs HATF	121	155	$\geq 0.2$	Not rejected

fuzzy rule selection is called Product-1-ALL TUN. In [2], Antonelli et al. proposed a multi-objective evolutionary algorithm for designing FRBCs namely PAES-RCS which train the rule bases concurrently with Rule Condition Selection (RCS). The membership function parameter values of linguistic values are tuned concurrently in the fuzzy rule and the Rule Condition Selection (RCS) process.

The experimental results of our proposed classifier HACO and the existing classifiers Product-1-ALL TUN, PAES-RCS and FURIA are shown in Table 6. It is seen that HACO has better classification accuracies on testing

sets than Product-1-ALL TUN, PAES-RCS and FURIA on 23, 21 and 18 of 23 experimented datasets, respectively. On the mean values, HACO has the highest classification accuracies (82.95 % in comparison with 80.57 %, 80.66 % and 80.34 %, respectively). On the mean value of model complexity, HACO has lowest model complexity (112.73 in comparison with 163.40, 355.23 and 281.49, respectively).

The results of the equivalent hypothesis tests on classification accuracies and model complexities of HACO, Product-1-ALL TUN, PAES-RCS and FURIA are shown in Table 7 and Table 8, respectively. Because all  $p$ -values in those tables are less than  $\alpha = 0.05$ , HACO classifier is better than the existing fuzzy set theory based classifiers namely PAES-RCS, Product-1-ALL TUN and FURIA on both testing accuracy and model complexity based measures.

## 6 Conclusion

A new Hedge Algebras based classifier design method with the application of co-optimization PSO algorithm for optimizing concurrently semantic parameter values

**Table 6** The experimental results and comparison between HACO and the existing fuzzy set theory based classifiers

No.	Dataset name	HACO		PAES-RCS		$\neq R \times C$	$\neq P_{te}$	Product-1-ALL TUN		$\neq R \times C$	$\neq P_{te}$	FURIA		$\neq R \times C$	$\neq P_{te}$
		$\#R \times C$	$P_{te}$	$\#R \times C$	$P_{te}$			$\#R \times C$	$P_{te}$			$\#R \times C$	$P_{te}$		
1	Appendicitis	19.65	89.09	35.28	85.09	-15.63	4.0	20.89	87.3	-1.24	1.79	19.0	85.18	0.65	3.91
2	Australian	51.99	87.20	329.64	85.80	-277.65	1.40	62.43	85.65	-10.44	1.55	89.6	85.22	-37.61	1.98
3	Bands	56.40	73.0	756.00	67.56	-699.6	5.44	104.09	65.8	-47.69	7.2	535.15	64.65	-478.75	8.35
4	Bupa	226.95	73.22	256.20	68.67	-29.25	4.55	210.91	67.19	16.04	6.03	324.12	69.02	-97.17	4.2
5	Cleveland	465.55	62.12	1140.0	59.06	-674.45	3.06	1020.66	58.8	-555.11	3.32	134.67	56.2	330.88	5.92
6	Dermatology	229.41	94.96	389.40	95.43	-159.99	-0.47	185.28	94.48	44.13	0.48	303.88	95.24	-74.47	-0.28
7	Glass	357.75	73.07	487.90	72.13	-130.15	0.94	534.88	71.28	-177.13	1.79	474.81	72.41	-117.06	0.66
8	Haberman	9.81	77.50	202.41	72.65	-192.6	4.85	21.13	71.88	-11.32	5.62	22.04	75.44	-12.23	2.06
9	Hayes-roth	111.87	84.79	120.0	84.03	-8.13	0.76	158.52	78.88	-46.65	5.91	188.1	83.13	-76.23	1.66
10	Heart	97.44	84.94	300.30	83.21	-202.86	1.73	164.61	82.84	-67.17	2.1	193.64	80	-96.2	4.94
11	Hepatitis	20.63	89.15	300.30	83.21	-279.67	5.94	20.29	88.53	0.34	0.62	52.38	84.52	-31.75	4.63
12	Ionosphere	102.35	91.64	670.63	90.40	-568.28	1.24	86.75	90.79	15.6	0.85	372.68	91.75	-270.33	-0.11
13	Iris	22.97	98.0	69.84	95.33	-46.87	2.67	18.54	97.33	4.43	0.67	31.95	94.66	-8.98	3.34
14	Mammogr.	75.50	84.25	132.54	83.37	-57.04	0.88	106.74	80.49	-31.24	3.76	16.83	83.89	58.67	0.36
15	Newthyroid	46.78	95.70	97.75	95.35	-50.97	0.35	56.47	94.6	-9.69	1.1	100.82	96.3	-54.04	-0.6
16	Pima	76.23	77.22	270.64	74.66	-194.41	2.56	57.2	77.05	19.03	0.17	127.5	74.62	-51.27	2.6
17	Saheart	75.04	70.26	525.21	70.92	-450.17	-0.66	110.84	70.13	-35.8	0.13	50.88	69.69	24.16	0.57
18	Sonar	46.43	78.94	524.60	77.0	-478.17	1.94	47.59	78.9	-1.16	0.04	309.96	82.14	-263.53	-3.2
19	Tae	157.89	61.67	323.14	60.81	-165.25	0.86	215.92	60.78	-58.03	0.89	43.0	43.08	114.89	18.59
20	Vehicle	178.20	68.32	555.77	64.89	-377.57	3.43	382.12	66.16	-203.92	2.16	2125.97	71.52	-1947.77	-3.2
21	Wdbc	36.83	96.81	183.70	95.14	-146.87	1.67	44.27	94.9	-7.44	1.91	356.12	96.31	-319.29	0.5
22	Wine	44.20	99.05	170.94	93.98	-126.74	5.07	58.99	93.03	-14.79	6.02	80.0	96.6	-35.8	2.45
23	Wisconsin	83.01	96.99	328.02	96.46	-245.01	0.53	69.11	96.35	13.9	0.64	521.1	96.35	-438.09	0.64
Mean		112.73	82.95	355.23	80.66			163.40	80.57			281.49	80.34		

**Table 7** The equivalent hypothesis test results on classification accuracies of HACO, PAES-RCS, Product-1-ALL TUN and FURIA.

VS	$R^+$	$R^-$	E. $P$ -value	Hypothesis
HACO vs PAES-RCS	270	6	3.338E-6	Rejected
HACO vs Product-1-ALL TUN	276	0	2.384E-7	Rejected
HACO vs FURIA	238	38	0.0014584	Rejected

**Table 8** The equivalent hypothesis test results on model complexities of HACO, PAES-RCS, Product-1-ALL TUN and FURIA.

VS	$R^+$	$R^-$	E. $P$ -value	Hypothesis
HACO vs PAES-RCS	276	0	2.384E-7	Rejected
HACO vs Product-1-ALL TUN	210	66	0.02768	Rejected
HACO vs FURIA	226	50	0.006032	Rejected

## References

- [1] Alcalá, R., Nojima, Y., Herrera, F., Ishibuchi, H. "Multiobjective genetic fuzzy rule selection of single granularity-based fuzzy classification rules and its interaction with the lateral tuning of membership functions", *Soft Computing*, 15(12), pp. 2303–2318, 2011. <https://doi.org/10.1007/s00500-010-0671-2>
- [2] Antonelli, M., Ducange, P., Marcelloni, F. "A fast and efficient multi-objective evolutionary learning scheme for fuzzy rule-based classifiers", *Information Sciences*, 283, pp. 36–54, 2014. <https://doi.org/10.1016/j.ins.2014.06.014>
- [3] Elkan, M., Galar, M., Sanz, J., Bustince, H. "CHI-BD: A fuzzy rule-based classification system for Big Data classification problems", *Fuzzy Sets and Systems*, 348, pp. 75–101, 2018. <https://doi.org/10.1016/j.fss.2017.07.003>
- [4] Ishibuchi, H., Yamamoto, T. "Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining", *Fuzzy Sets and Systems*, 141(1), pp. 59–88, 2004. [https://doi.org/10.1016/S0165-0114\(03\)00114-3](https://doi.org/10.1016/S0165-0114(03)00114-3)
- [5] Ishibuchi, H., Yamamoto, T. "Rule weight specification in fuzzy rule-based classification systems", *IEEE Transactions on Fuzzy Systems*, 13(4), pp. 428–435, 2005. <https://doi.org/10.1109/TFUZZ.2004.841738>
- [6] Rey, M. I., Galende, M., Fuente, M. J. Sainz-Palmero, G. I. "Multi-objective based Fuzzy Rule Based Systems (FRBSs) for trade-off improvement in accuracy and interpretability: A rule relevance point of view", *Knowledge-Based Systems*, 127, pp. 67–84, 2017. <https://doi.org/10.1016/j.knsys.2016.12.028>
- [7] Cat Ho, N., Wechler, W. "Hedge algebras: An algebraic approach to structure of sets of linguistic truth values", *Fuzzy Sets and Systems*, 35(3), pp. 281–293, 1990. [https://doi.org/10.1016/0165-0114\(90\)90002-N](https://doi.org/10.1016/0165-0114(90)90002-N)
- [8] Ho, N. C., Wechler, W. "Extended hedge algebras and their application to fuzzy logic", *Fuzzy Sets and Systems*, 52(3), pp. 259–281, 1992. [https://doi.org/10.1016/0165-0114\(92\)90237-X](https://doi.org/10.1016/0165-0114(92)90237-X)
- [9] Ho, N. C., Long, N. V. "Fuzziness measure on complete hedges algebras and quantifying semantics of terms in linear hedge algebras", *Fuzzy Sets and Systems*, 158(4), pp. 452–471, 2007. <https://doi.org/10.1016/j.fss.2006.10.023>
- [10] Bui, H.-L., Le, T.-A., Bui, V.-B. "Explicit formula of hedge-algebras-based fuzzy controller and applications in structural vibration control", *Applied Soft Computing*, 60, pp. 150–166, 2017. <https://doi.org/10.1016/j.asoc.2017.06.045>
- [11] Nguyen, C. H., Pedrycz, W., Duong, T. L., Tran, T. S. "A genetic design of linguistic terms for fuzzy rule based classifiers", *International Journal of Approximate Reasoning*, 54(1), pp. 1–21, 2013. <https://doi.org/10.1016/j.ijar.2012.07.007>
- [12] Nguyen, C. H., Tran, T. S., Pham, D. P. "Modeling of a semantics core of linguistic terms based on an extension of hedge algebra semantics and its application", *Knowledge-Based Systems*, 67, pp. 244–262, 2014. <https://doi.org/10.1016/j.knsys.2014.04.047>
- [13] Nguyen, C. H., Hoang, V. T., Nguyen, V. L. "A discussion on interpretability of linguistic rule based systems and its application to solve regression problems", *Knowledge-Based Systems*, 88, pp. 107–133, 2015. <https://doi.org/10.1016/j.knsys.2015.08.002>
- [14] Anh, N. T., Son, T. T. "Partition Fuzzy Domain with Multi-granularity Representation of Data Basing on Hedge Algebra Approach", *Journal of Computer Science and Cybernetics*, 34(1), pp. 63–76, 2018. <https://doi.org/10.15625/1813-9663/34/1/10797>
- [15] Ngo, H. H., Nguyen, C. H., Nguyen, V. Q. "Multichannel image contrast enhancement based on linguistic rule-based intensifiers", *Applied Soft Computing Journal*, 76, pp. 744–762, 2019. <https://doi.org/10.1016/j.asoc.2018.12.034>
- [16] Long, D. T. "A Genetic Algorithm Based Method for Timetabling Problems Using Linguistics of Hedge Algebra in Constraints", *Journal of Computer Science and Cybernetics*, 32(4), pp. 285–301, 2016. <https://doi.org/10.15625/1813-9663/32/4/7962>
- [17] Dinh, P. P., Cat, H. N., Thanh, T. N. "Multi-objective particle swarm optimization algorithm and its application to the fuzzy rule based classifier design problem with the order based semantics of linguistic terms", In: *The 2013 RIVF International Conference on Computing and Communication Technologies – Research, Innovation, and Vision for Future (RIVF)*, Hanoi, Vietnam, 2013, pp. 12–17. <https://doi.org/10.1109/RIVF.2013.6719858>

and fuzzy rule systems instead of two phase optimization of the existing Hedge Algebras based Classifier Design Methods is presented in this paper. The results of experiments executed over 23 real-world datasets have shown that our proposed classifier is better than the existing ones. Our proposed co-optimization PSO algorithm can be parallelized to improve the running performance.

## Acknowledgement

The authors gratefully acknowledge use of service and facilities of University of Transport and Communications, Vietnam.

- [18] Kennedy, J., Eberhart, R. "Particle swarm optimization", In: Proceedings of ICNN'95 – International Conference on Neural Networks, vol. 4., IEEE, Piscataway, NJ, USA, 1995, pp. 1942–1948.  
<https://doi.org/10.1109/icnn.1995.488968>
- [19] Eberhart, R. , Kennedy, J. "A new optimizer using particle swarm theory", In: Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 1995, pp. 39–43.  
<https://doi.org/10.1109/MHS.1995.494215>
- [20] Lechuga, M. S. "Multi-Objective Optimisation using Sharing in Swarm Optimisation Algorithms", Doctor Thesis, The University of Birmingham, 2006.
- [21] Alcalá-Fdez, J., Fernandez, A., Luengo, J., Derrac, J., García, S., Sánchez, L., Herrera. F. "KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework", Journal of Multiple-Valued Logic and Soft Computing, 17(2–3), pp. 255–287, 2011. [online] Available at: <http://sci2s.ugr.es/keel/datasets.php> [Accessed: 07 April 2020]
- [22] Demšar, J. "Statistical Comparisons of Classifiers over Multiple Data Sets", The Journal of Machine Learning Research, 7, pp. 1–30, 2006. [online] Available at: <https://dl.acm.org/doi/10.5555/1248547.1248548> [Accessed: 07 April 2020]