International Journal of Approximate Reasoning 119 (2020) 20–39

Contents lists available at ScienceDirect

International Journal of Approximate Reasoning

www.elsevier.com/locate/ijar

# A new rule reduction and training method for extended belief rule base based on DBSCAN algorithm

An Zhang [a,b], Fei Gao [a], Mi Yang [a], Wenhao Bi [a,*]

[a] School of Aeronautics, Northwestern Polytechnical University, Xi'an, Shaanxi, China
[b] CETC Key Laboratory of Data Link Technology, Xi'an, Shaanxi, China

ARTICLE INFO

*Article history:*
Received 12 November 2018
Received in revised form 11 December 2019
Accepted 23 December 2019
Available online 3 January 2020

*Keywords:*
Extended belief rule base
Rule reduction
Parameter learning

ABSTRACT

Rule reduction is one of the focuses of numerous researches on belief-rule-based system, in some cases, too many redundant rules may be a concern to the rule-based system. Though rule reduction methods have been widely used in the belief-rule-based system, extended belief-rule-based system, which is an expansion of belief-rule-based system, still lacks methods to reduce and train rules in the extended belief rule base (EBRB). To this end, this paper proposes an EBRB reduction and training method. Based on the density-based spatial clustering applications with noise (DBSCAN) algorithm, a new EBRB reduction method is proposed, where all the rules in the EBRB will be visited and rules within the distance of the fusion threshold will be fused. Moreover, the EBRB training method using parameter learning, which uses a set of training data to train the parameters of EBRB, is also proposed to improve the accuracy of the EBRB system. Two case studies of regression and classification are used to illustrate the feasibility and efficiency of the proposed EBRB reduction and training method. Comparison results show that the proposed method can effectively downsize the EBRB and increase the accuracy of EBRB system.



## 1. Introduction

Among various knowledge representation schemes, rule-based system has been widely recognized as one of the most commonly used frameworks due to its advantages of expressing various types of knowledge under the same framework [1]. Constructed by human knowledge in the forms of IF-THEN rules, it has become one of the fastest-growing methods in the field of decision support system and artificial intelligence [2,3]. Normally, simple IF-THEN rules such as "IF **failure rate** is **high**, THEN **safety estimate** is **poor**" are used to construct the rule base, which is an essential part of the rule-based system, and it has attracted numerous studies on its optimization [4–6]. Among those researches, rule base reduction has always been a focus as excessive rules could decrease the accuracy and increase the complexity of the rule-based system [7–9]. Moreover, as both the antecedent and the consequent terms of the IF-THEN rule are believed to be 100% certain, while such strict knowledge representation scheme may be inefficient in expressing information with uncertainty, many new rule-based systems have been developed to enhance the ability to deal with both qualitative and quantitative information under uncertainty.

In this regard, Yang et al. [10] proposed the belief-rule-base methodology using the evidential reasoning (RIMER) approach in 2006, which is developed on the basis of Dempster-Shafer theory of evidence [11–13], Bayesian probability theory

* Corresponding author.
*E-mail address:* biwenhao@nwpu.edu.cn (W. Bi).

https://doi.org/10.1016/j.ijar.2019.12.016

[14], fuzzy set theory [15,16] and traditional IF-THEN rule-based system [1,17]. This methodology has its advantages in reflecting information under conditions of uncertainty, such as fuzziness, ignorance, and incompleteness, and has been widely used in consumer preference prediction [18–20], medical diagnosis [21–23], risk analysis [24–27], and many other fields. Compared with belief functions [13,17], the belief rule-based system is more suitable for problems involving prior knowledge, such as regression and classification problems. By grouping belief rules into one belief rule base (BRB), the belief rule-based system can effectively address different decision problems, however, as BRB is an essential part of the belief rule-based system and has a significant impact on the reasoning results, its optimization method has attracted extensive studies [28–30]. Moreover, belief rule base reduction has also become a focus of researchers, and many methods have been applied to provide promising belief rule base reduction methods [31–34]. Furthermore, parameter learning method combined with belief rule base reduction has been studied by some researchers to optimize the belief rule base and reduce its scale at the same time [35]. Recently, based on the RIMER approach, Liu et al. [36] proposed a more general belief rule-based system by introducing belief structures into the antecedent attributes of each rule, called the extended belief rule base (EBRB), to better deal with uncertainty in the antecedent attributes. With belief degrees embedded in the entire antecedent attributes of each rule, EBRB can more effectively deal with different kinds of uncertainty and present more accurate results, and has been used in fields such as health estimation [37], sensor configuration optimization [38], activity recognition [39] and some other fields [40,41].

However, several challenges still need to be addressed in the EBRB system. The first is to reduce the size of EBRB system and avoid inconsistency, and the second challenge is to enhance its representation power, which is essential for rule-based systems.

For the first challenge, some researchers approach it by adjusting the number of activated rules in EBRB system, which could reduce the inference time [42–44]. Calzada et al. [43] proposed a dynamic rule activation (DRA) method for EBRB system, where the activated rules are selected dynamically to search for a balance between the incompleteness and inconsistency. Lin et al. [44] proposed the rule activation method based on VP-tree and MVP-tree, where only partial rules will be retrieved and visited while calculating each rule's activation weight. Yang et al. [45] proposed the consistency analysis-based rule activation (CABRA) method to redefine suitable activated rules by using the set of consistent rules as an activation framework, which enables the EBRB system to overcome problems of inconsistency and incompleteness. Yang et al. [41] constructed the activation region of extended belief rules and revised the calculation formula of activation weights to redefine activated rules, and proposed new activation rule determination and weight calculation procedures. However, these rule activation methods cannot effectively reduce the size of EBRB, and information could be lost as only limited rules are activated for a given input. As for EBRB reduction, there have been few studies, Yang et al. [46] proposed EBRB rule reduction method based on data envelopment analysis (DEA), where DEA is used to evaluate the efficiency of each rule and remove inefficient rules from EBRB to reduce its size. As explained in Yang et al. [46], EBRB reduction is more important and effective than rule activation to solve the first challenge since: (1) the size of EBRB could be overlarge without EBRB reduction as input-output data pair can be easily transferred into belief rules in EBRB; (2) activating all the rules in EBRB for any input data is the exact reason for inconsistency, and it could get worse when the size of EBRB is increasing; (3) simply activate certain rules for each input data could lead to the loss of information as several related rules could be ignored.

For the second challenge, one of the most common ways is to increase the number of rules in the rule base, which may also increase its time cost. Recently, rule base training has been used as an efficient way to improve the reasoning performance of belief rule-based system. Yang et al. [47] proposed a new activation weight calculation and parameter optimization method based on sensitivity analysis, by using the new parameter optimization method, it enhanced the accuracy of the EBRB system. Though EBRB uses training data for rule generation, the training process is still necessary for increasing its representation power and enhancing its performance.

In order to address the abovementioned challenges, density-based spatial clustering of applications with noise (DBSCAN) algorithm and parameter learning are applied for EBRB reduction and training in this paper. For the first challenge, a new EBRB reduction method is proposed, where similar belief rules would be searched and fused using the DBSCAN algorithm based on the Euclidean distance between different rules. For the second challenge, the parameters of EBRB, namely, belief degrees of antecedent attributes, consequent belief degrees, attribute weights and rule weights are trained using an extended parameter learning method to enhance its accuracy. For an oversize EBRB, EBRB reduction method will be firstly used to prune the belief rules, and EBRB training method will be applied to train the parameter of the newly constructed EBRB. The main advantage of the proposed method is that it can effectively reduce the size of EBRB while maintaining relatively good accuracy. To illustrate the effectiveness of the proposed method, two cases, including function approximation and classification, are applied. Two main aspects, namely, accuracy and the number of rules, are used to compare with other EBRB systems.

The rest of this paper is organized as follows. Section 2 briefly reviews the basics of evidential reasoning (ER) approach and EBRB system. Section 3 proposes the EBRB training method and the EBRB reduction method. Section 4 provides two case studies to demonstrate the efficiency of the proposed EBRB reduction and training method, and the paper is concluded in Section 5.

## 2. Preliminaries

### 2.1. Evidential reasoning approach

To capture incomplete, fuzzy and ignorant information, Yang and Singh [48] proposed the evidential reasoning approach based on the D-S theory of evidence. It uses belief structure to represent an assessment of an attribute $e_i$ as follows [49]:

$$S(e_i) = \{(H_1, \beta_{1,i}), \dots, (H_n, \beta_{n,i}), \dots, (H_N, \beta_{N,i}), n = 1, \dots, N\} \tag{1}$$

where $H_n$ is the $n$th evaluation grade, $\beta_{n,i}$ denotes a degree of belief, $\beta_{n,i} \geq 0$, $\sum_{n=1}^{N} \beta_{n,i} \leq 1$. The above belief structure reads that the attribute $e_i$ is assessed to $H_n$ with a degree of belief $\beta_{n,i}$. An assessment is complete when $\sum_{n=1}^{N} \beta_{n,i} = 1$, and is incomplete when $\sum_{n=1}^{N} \beta_{n,i} < 1$.

Let $\beta_n$ be the degree of belief to which the overall attribute is assessed to $H_n$. Assume there are $L$ attributes $\boldsymbol{e} = \{e_1, \dots, e_i, \dots, e_L\}$ with $N$ grades to assess the general attribute, then the ER analytical algorithm [50], which has been equivalently transformed from the ER recursive algorithm [49], calculates the overall combined belief degree $\beta_n$ in $H_n$ as follows:

$$\beta_n = \frac{d[\prod_{i=1}^{L} (\omega_i \beta_{n,i} + 1 - \omega_i \sum_{k=1}^{N} \beta_{k,i}) - \prod_{i=1}^{L} (1 - \omega_k \sum_{k=1}^{N} \beta_{k,i})]}{1 - d[\prod_{i=1}^{L} (1 - \omega_i)]} \tag{2}$$

with

$$d = [\sum_{n=1}^{N} \prod_{i=1}^{L} (\omega_i + 1 - \omega_i \sum_{k=1}^{N} \beta_{k,i}) - (N-1) \prod_{i=1}^{L} (1 - \omega_i \sum_{n=1}^{N} \beta_{n,i})]^{-1} \tag{3}$$

where $\beta_{n,i}$ is the degree of belief to which the $i$th attribute $e_i$ is assessed to $H_n$, $\omega_i$ is the weight of the $i$th attribute, and it satisfies $0 \leq \omega_i \leq 1$. $\beta_n$ is the combined overall degree of belief in $H_n$, and it satisfies $\beta_n \geq 0$, $\sum_{n=1}^{N} \beta_n \leq 1$.

### 2.2. Extended belief rule base

An EBRB (Extended Belief Rule Base) is one of the two main components of an EBRB system, and is designed with belief structure embedded in the consequent terms as well as in the antecedent attributes of each rule, which consists of $M$ antecedent attributes and one consequent attribute. Each antecedent attribute $U_i$ $(i = 1, \dots, M)$ has $J_i$ referential values $A_{ij}$ $(j = 1, \dots, J_i)$, and the consequent $D$ has $N$ referential values $D_n$ $(n = 1, \dots, N)$. The $k$th extended belief rule $R_k$ $(k = 1, \dots, L)$ can be expressed as follows:

$$\begin{aligned} R_k :& \text{IF } U_1 \text{ is } \{(A_{1,j}^k, \alpha_{1,j}^k), j = 1, \dots, J_1\} \wedge \dots \wedge U_M \text{ is } \{(A_{M,j}^k, \alpha_{M,j}^k), j = 1, \dots, J_M\}, \\ & \text{THEN } D \text{ is } \{(D_n, \beta_n^k), n = 1, \dots, N\}, \text{ with rule weight } \theta_k \text{ and attribute weights } \{\delta_1, \dots, \delta_M\} \end{aligned} \tag{4}$$

where $\alpha_{i,j}^k$ is the belief degree to which $U_i$ is evaluated to be the referential value $A_{i,j}^k$ in the $k$th rule with $0 \leq \alpha_{i,j}^k \leq 1$ and $\sum_{j=1}^{J_i} \alpha_{i,j}^k \leq 1$ $(i = 1, \dots, M)$. $\beta_n^k$ is the belief degree to which $D$ is evaluated to be the referential value $D_n$ in the $k$th rule with $0 \leq \beta_n^k \leq 1$ and $\sum_{n=1}^{N} \beta_n^k \leq 1$. The consequent is complete when $\sum_{n=1}^{N} \beta_n^k = 1$, otherwise, it is incomplete. $\theta_k$ $(0 \leq \theta_k \leq 1)$ and $\delta_i$ $(0 \leq \delta_i \leq 1)$ denotes the rule weight and attribute weight respectively. $k = 1, \dots, L$, and $L$ is the number of the extended belief rules in the EBRB.

### 2.3. Inference method

The inference process of the EBRB system comprises the calculation of the individual matching degree, the calculation of the activation weight and combination of the activated rule. Firstly, suppose $x_i$ represents the input of the $i$th antecedent attribute $U_i$, then it can be transformed into belief structure using utility-based information transformation technique [10] as follows:

$$S(x_i) = \{(A_{i,j}, \alpha_{i,j}), j = 1, \dots, J_i\} \tag{5}$$

with

$$\begin{aligned} &\alpha_{i,j} = \frac{u(A_{i,j+1}) - x_i}{u(A_{i,j+1}) - u(A_{i,j})}, \alpha_{i,j+1} = 1 - \alpha_{i,j}, \text{ if } u(A_{i,j}) \leq x_i \leq u(A_{i,j+1}) \\ &\alpha_{i,m} = 0, \text{ if } m = \{1, \dots, J_i\} \text{ and } m \neq j, j+1 \end{aligned} \tag{6}$$

where $A_{i,j}$ represents the $j$th referential value in the $i$th antecedent attribute, $\alpha_{i,j}$ is the belief degree to which the input $x_i$ is assessed to the referential value $A_{i,j}$, $u(A_{ij})$ is the utility value of $A_{i,j}$.

**Step 1: Calculate the individual matching degree**

Once the input is obtained as $S(x_i) = \{(A_{i,j}, \alpha_{i,j}), j = 1, \ldots, J_i\}$, then the individual matching degree $S_i^k$ of $x_i$ to $U_i$ of the $k$th rule, which represents how *close* these two belief structures are, can be calculated based on the similarity measure as follows:

$$S_i^k = S^k(x_i, U_i) = \begin{cases} 1 - d_i^k, & d_i^k \le 1 \\ 0, & d_i^k > 1 \end{cases} \tag{7}$$

where $d_i^k$ denotes the distance between two belief structures. Since they are actually probability distributions (either an objective probability, e.g., relative frequency, or a subjective probability, e.g., confidence or belief), $d_i^k$ can be best summarized using Euclidean distance as follows [36]:

$$d_i^k = d^k(x_i, U_i) = \sqrt{\sum_{j=1}^{J_i} (\alpha_{i,j} - \alpha_{i,j}^k)^2} \tag{8}$$

**Remark 1.** Eq. (7) is a modification of the individual matching degree calculation formula proposed in Liu et al. [36], which only focuses on cases where $d_i^k \in [0, 1]$. However, as Eq. (8) illustrates, there are cases where $d_i^k > 1$, and based on the individual matching degree calculation formula [36], the individual matching degree $S_i^k$ would be less than 0 correspondingly, which exceeds its defined interval. As it can be seen that there is no similarity between the two belief structures, the individual matching degree should be 0.

**Remark 2.** The similarity between two belief structures $S_1$, $S_2$ represents how *close* the two belief structures are. They are regarded as completely similar when two belief structures are identical, i.e. $S_1 = S_2$. When two belief structures have no intersection, i.e. $S_1 \cap S_2 = \varnothing$, it is regarded as there is no similarity between them. As the distance is defined as $d_i^k \in [0, \sqrt{2}]$, the similarity measure is defined as $S_i^k = 1 - d_i^k$ rather than $S_i^k = 1/(1 + d_i^k)$ to maintain generality [36]. However, as shown above, the similarity measure is not strictly defined, and it may change in accordance with the application situation.

**Remark 3.** Other distance measure may be applied to calculate the distance between the input and the extended belief rule as it is not strictly defined [36]. However, in this case, other distance measures such as Manhattan distance, Chebyshev distance, and Mahalanobis distance cannot effectively or accurately reflect the distance between two belief structures. Among them, Mahalanobis distance clearly cannot be used for this problem since the distance shouldn't be dependent on the distribution of data. As for Manhattan distance and Chebyshev distance, considering two belief structures $S_1 = \{(H_1, 0), (H_2, 0.2), (H_3, 0.8), (H_4, 0), (H_5, 0)\}$, $S_2 = \{(H_1, 0.1), (H_2, 0.3), (H_3, 0.6), (H_4, 0), (H_5, 0)\}$, with Euclidean distance $d_E = 0.2449$. Using Manhattan distance, $d_M = \sum_{n=1}^{N} |\beta_{1,n} - \beta_{2,n}| = 0.4$, which doesn't represent the direct distance of the two belief structures and enlarges the actual difference. Using Chebyshev distance, $d_C = \max_{n=1,\ldots,N} |\beta_{1,n} - \beta_{2,n}| = 0.2$, which only focuses on the biggest absolute difference between two belief structures, and disregards the difference on other attributes. Thus, Euclidean distance could more accurately measure the distance between two belief structures in this case. More options about distance measure can be used depending on the application context, for more information, refer to [36,41,45,51].

**Example 1.** (Calculation of the individual matching degree): Suppose the belief structure of the antecedent **consequence severity** of an extended belief rule is

$$S(U) = \{(negligible, 0), (marginal, 0), (moderate, 0), (critical, 0.3), (catastrophic, 0.7)\}$$

and the input is

$$S(x) = \{(negligible, 0), (marginal, 0), (moderate, 1), (critical, 0), (catastrophic, 0)\}$$

In this case, belief structures of the antecedent attribute and the input are completely different from each other as they support totally different assessments, i.e. $S(U) \cap S(x) = \varnothing$, and can be regarded as *no similarity*. The Euclidean distance between the input and the belief rule can be calculated as $d_i^k = \sqrt{1^2 + 0.3^2 + 0.7^2} = 1.2570$, thus, using Eq. (7), the similarity measure can be obtained as $S_i^k = 0$, which is in line with the facts.

**Step 2: Calculate the activation weight**

The activation weight for the $k$th extended belief rule can be calculated as follows:

$$\omega_k = \frac{\theta_k \prod_{i=1}^{M} (S_i^k)^{\overline{\delta}_i}}{\sum_{l=1}^{L} (\theta_l \prod_{i=1}^{M} (S_i^l)^{\overline{\delta}_i})} \tag{9}$$

with

$$\overline{\delta}_i = \frac{\delta_i}{\max_{i=1,\ldots,M}\{\delta_i\}} \tag{10}$$

where $\theta_k$ and $\delta_i$ are given in the EBRB, it should be noted that

$$0 \le \omega_k \le 1 (k = 1, \ldots, L), \quad \sum_{k=1}^{L} \omega_k = 1 \tag{11}$$

with $\omega_k = 0$ denoting that the $k$th rule is not activated.

**Step 3: Aggregate the activated rules using ER**

After calculating the activation weight, all the activated rules should be aggregated using the evidential reasoning (ER) algorithm. Using Eq. (2), the combined belief degree $\beta_n$ can be calculated as follows:

$$\beta_n = \frac{d[\prod_{k=1}^{L} (\omega_k \beta_n^k + 1 - \omega_k \sum_{n=1}^{N} \beta_n^k) - \prod_{k=1}^{L} (1 - \omega_k \sum_{i=1}^{N} \beta_i^k)]}{1 - d[\prod_{k=1}^{L} (1 - \omega_k)]} \tag{12}$$

with

$$d = [\sum_{n=1}^{N} \prod_{k=1}^{L} (\omega_k \beta_n^k + 1 - \omega_k \sum_{j=1}^{N} \beta_j^k) - (N-1) \prod_{k=1}^{L} (1 - \omega_k \sum_{j=1}^{N} \beta_j^k)]^{-1} \tag{13}$$

where $\beta_n^k$ is the degree of belief of the $n$th consequent in the $k$th rule, $\omega_k$ is the activation weight of the $k$th rule, and $\beta_n$ is the combined belief degree, which satisfies $\beta_n \ge 0$ and $\sum_{n=1}^{N} \beta_n \le 1$.

For regression problem, suppose $u(D_n)$ is the utility value of the $n$th consequent grade, and $\beta_n$ is the corresponding belief degree, then the inference output of the EBRB system can be obtained as follows:

$$y = \sum_{n=1}^{N} u(D_n)\beta_n + \frac{u(D_1) + u(D_n)}{2}(1 - \sum_{n=1}^{N} \beta_n) \tag{14}$$

For classification problem, suppose that $D_n$ represents the $n$th class, then the inference result can be expressed as follows:

$$f = D_n, n = \mathrm{argmax}_{n=1,\ldots,N}(\beta_n) \tag{15}$$

## 3. EBRB reduction and training method

### 3.1. EBRB training method using parameter learning

Yang et al. [29] proposed optimal learning model for training BRBs, where rule weight $\theta_k$, attribute weight $\delta_i$ and consequent belief degree $\beta_n^k$ are trained simultaneously using a set of input-output data pairs to enhance BRB performance. However, as for the EBRB system, because belief degrees are embedded in all the antecedent terms of each rule as well, the training parameters should be expanded to antecedent attribute belief degree $\alpha_{i,j}^k$, consequent belief degree $\beta_n^k$, rule weight $\theta_k$ and attribute weight $\delta_i$. The EBRB training process is shown in Fig. 1, and the parameter learning problem can be represented as the following nonlinear optimization problem:

$$\begin{aligned} &\min \ f(\mathbf{P}) \\ &\text{s.t. } A(\mathbf{P}) = 0, \quad B(\mathbf{P}) \ge 0 \end{aligned} \tag{16}$$

where $f(\mathbf{P})$ is the objective function; $\mathbf{P}$ is the training parameter vector, including $\alpha_{ij}^k$, $\beta_n^k$, $\theta_k$, and $\delta_i$; $A(\mathbf{P})$ is the equality constraint functions; $B(\mathbf{P})$ is the inequality constraint functions.

In the parameter learning process, a set of observations on the system inputs and outputs is required. In the following, assume $(x_t, y_t), t = 1, \ldots, T$ is a set of system input-output data pairs, where $x_t$ is the input and $y_t$ is the corresponding output, and both $x$ and $y$ are numerical data. Suppose that a set of training data is provided in the form of $T$ input-output

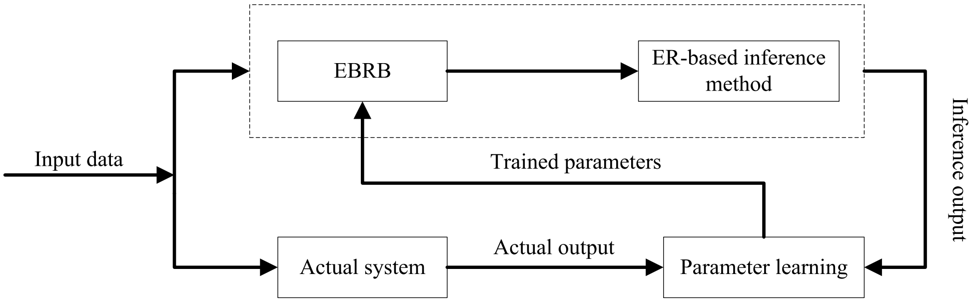


**Fig. 1.** EBRB training process.

pairs $(\hat{x}_t, \hat{y}_t)$, with $\hat{y}_t$ being a numerical value. The output of the EBRB system is calculated using Eq. (14), where the utility value of an individual consequent $D_n$ is denoted by $u(D_n)$.

Since both the output in the training data set and the output of the EBRB system are numerical, the objective of the optimization model is to determine the antecedent attribute belief degrees $\alpha_{i,j}^k$, the consequent belief degrees $\beta_n^k$, the rule weights $\theta_k$ and the attribute weights $\delta_i$, which can be denoted as $\mathbf{P} = (\alpha_{i,j}^k, \beta_n^k, \theta_k, \delta_i)$, in order to minimize the objective function, which is assumed to be the total mean squared error (MSE) in this paper and is defined as follows:

$$\min_{\mathbf{P}} \{\xi(\mathbf{P})\} \tag{17}$$

where $\xi(\mathbf{P}) = \frac{1}{T}\sum_{t=1}^{T}(y_t - \hat{y}_t)^2$ is the total mean squared error, $y_t = \sum_{n=1}^{N} u(D_n)\beta_n(t) + \frac{u(D_1)+u(D_N)}{2}(1 - \sum_{n=1}^{N}\beta_n)$ is the expected output of the EBRB system for the $t$th input. $T$ is the number of input-output pairs, $\hat{y}_t$ is the observed output, and $(y_t - \hat{y}_t)$ is the error of the $t$th input-output data pair.

For an EBRB system, the constraints of the parameter learning model in Eq. (16) are given as follows:

(1) A belief degree must not be less than zero or more than one, i.e.,

$$\begin{aligned} 0 \le \alpha_{i,j}^k \le 1 \\ 0 \le \beta_n^k \le 1 \end{aligned} \tag{18}$$

(2) If the $i$th antecedent attribute assessment of the $k$th rule is complete, its total belief degree should be equal to one, i.e.,

$$\sum_{j=1}^{J_i} \alpha_{i,j}^k = 1 \tag{19}$$

Otherwise, it should be less than one.

(3) Suppose the actual value of the $i$th antecedent attribute of the $k$th rule is $x_i$, then the expected utility value of the $i$th antecedent attribute of the $k$th rule should be equal to $x_i$, i.e.,

$$\sum_{j=1}^{J_i} u(A_{i,j}^k)\alpha_{i,j}^k = x_i \tag{20}$$

(4) If the consequent assessment of the $k$th rule is complete, its total belief degree should be equal to one, i.e.,

$$\sum_{n=1}^{N} \beta_n^k = 1 \tag{21}$$

Otherwise, it should be less than one.

(5) The rule weight is normalized, thus, it is between zero and one, i.e.,

$$0 \le \theta_k \le 1 \tag{22}$$

(6) The attribute weight is normalized, thus, it is between zero and one, i.e.,

$$0 \le \delta_i \le 1 \tag{23}$$

**Table 1**
Initial EBRB of Example 2.

| Attribute weight | Rule | Rule weight | $x$ | $f(x)$ | Antecedent attribute | Consequent |
|---|---|---|---|---|---|---|
| 1.0000 | 1 | 1.0000 | 0.0000 | 0.0000 | (1.0000, 0.0000, 0.0000) | (1.0000, 0.0000, 0.0000, 0.0000, 0.0000) |
| | 2 | 1.0000 | 0.5000 | 0.2397 | (0.5000, 0.5000, 0.0000) | (0.5206, 0.4794, 0.0000, 0.0000, 0.0000) |
| | 3 | 1.0000 | 1.0000 | 0.8415 | (0.0000, 1.0000, 0.0000) | (0.0000, 0.3171, 0.6829, 0.0000, 0.0000) |
| | 4 | 1.0000 | 1.5000 | 1.4962 | (0.0000, 0.5000, 0.5000) | (0.0000, 0.0000, 0.0075, 0.9925, 0.0000) |
| | 5 | 1.0000 | 2.0000 | 1.8186 | (0.0000, 0.0000, 1.0000) | (0.0000, 0.0000, 0.0000, 0.3628, 0.6372) |

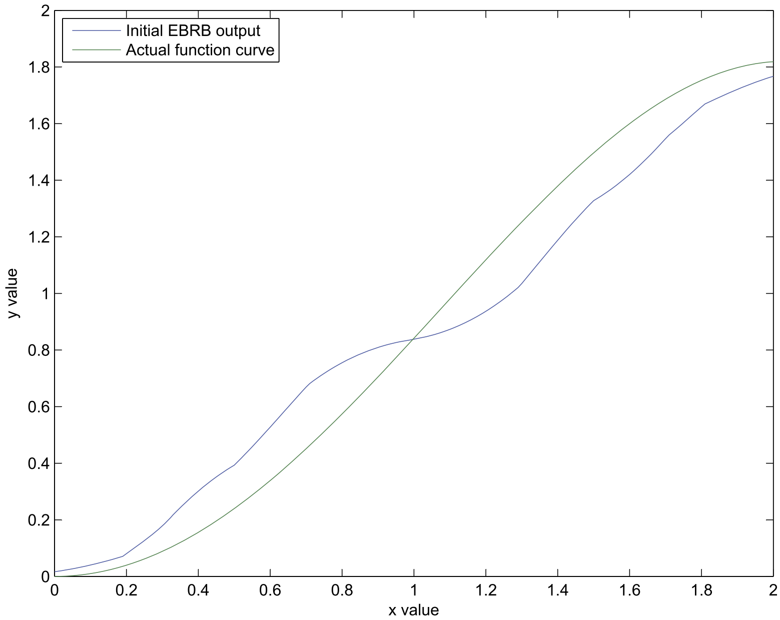


**Fig. 2.** Output of the initial EBRB system of Example 2.

With these been defined, the parameter learning problem can be further determined. The objective function defined in Eq. (17) is a nonlinear function of the training parameters. With additional parameters and constraints in the EBRB systems, the parameter learning problem can be treated as a large-scale nonlinear optimization problem. Active-set method has been widely used to the aforementioned problem, the goal of this method is to predict an active set, in which the set of constraints are satisfied with equality at the solution of the problem [35]. Such a method can be achieved using existing optimization software packages, such as the Matlab Optimization Toolbox.

**Remark 4.** Though the EBRB generation method uses training data to generate belief rules with no time-consuming iterative procedure and can achieve relatively satisfactory performance compared with BRB. However, with the removal of the training process, its reasoning performance could be affected, and it is necessary to train the EBRB to enhance its representing power.

In order to illustrate the effect that the training process could have on the performance of EBRB system, the following example is given.

**Example 2.** (Analysis of the influence of EBRB training): Consider the following nonlinear function:

$$f(x) = x * \sin(x) \tag{24}$$

where the input $x$ is bounded in the interval $[0, 2]$, and the referential value of three antecedent grades and five consequent grades are defined as follows:

$$U = \{0, 1, 2\}, \qquad D = \{0, 0.5, 1, 1.5, 2\} \tag{25}$$

A set of five input-output pairs selected evenly from the range of values for $x$ is used to construct the initial EBRB and also used as the training set. The initial EBRB is shown in Table 1, and its output curve is shown in Fig. 2.

**Table 2**
Trained EBRB of Example 2.

| Attribute weight | Rule | Rule weight | $x$ | $f(x)$ | Antecedent attribute | Consequent |
|---|---|---|---|---|---|---|
| 1.0000 | 1 | 1.0000 | 0.0000 | 0.0000 | (1.0000, 0.0000, 0.0000) | (1.0000, 0.0000, 0.0000, 0.0000, 0.0000) |
| | 2 | 1.0000 | 0.5000 | 0.2397 | (0.5081, 0.4838, 0.0081) | (0.9133, 0.0867, 0.0000, 0.0000, 0.0000) |
| | 3 | 0.5872 | 1.0000 | 0.8415 | (0.0000, 1.0000, 0.0000) | (0.0412, 0.2432, 0.6630, 0.0125, 0.0401) |
| | 4 | 0.9886 | 1.5000 | 1.4962 | (0.0299, 0.4402, 0.5299) | (0.0051, 0.0020, 0.0030, 0.5321, 0.4578) |
| | 5 | 0.9995 | 2.0000 | 1.8186 | (0.0000, 0.0000, 1.0000) | (0.0490, 0.0240, 0.0000, 0.1054, 0.8216) |

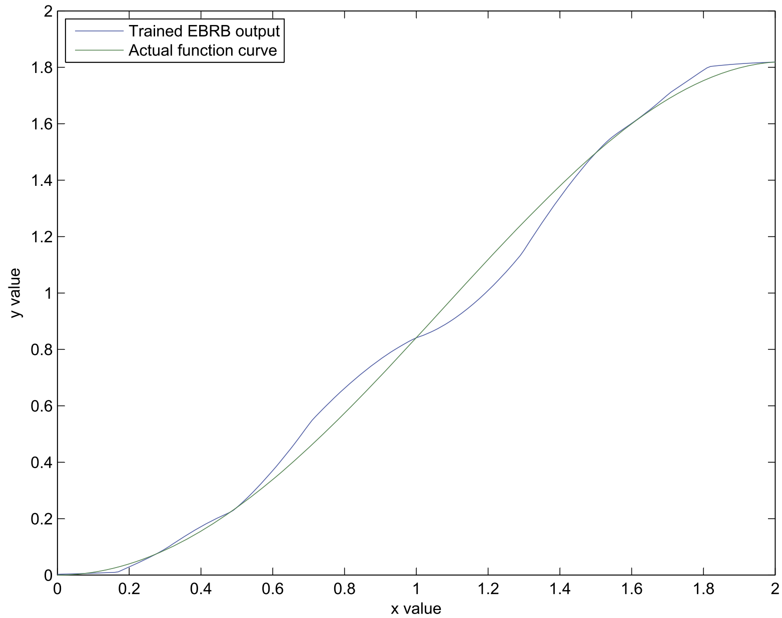


**Fig. 3.** Output of the trained EBRB system of Example 2.

As aforementioned, the same set of input-output data is used as the training set for EBRB training, and the trained EBRB is shown in Table 2, with its curve shown in Fig. 3

As shown in Fig. 2 and Fig. 3, though the EBRB generation method can generate relatively reliable initial EBRB based on the training data set, its reasoning performance is still limited, and using the same set as the training set, a more accurate result can be reached with the trained EBRB, with the MSE reduced from 0.0183 to 0.000006. Therefore, it is necessary to conduct EBRB training using the training data set to enhance the performance of EBRB system. Moreover, we can use part of the training data set to generate the initial EBRB, and using the whole training set to conduct EBRB training, which could reduce the size of initial EBRB while maintaining relatively good reasoning performance, as well as ensure the time cost is within the appropriate range.

**Example 3.** (EBRB training): Suppose there is a single-input system, the input $x$ is bounded in the interval $[0, 2]$, and the output is $f(x) = x * sin(x^2)$. To construct an EBRB, the referential value of three antecedent grades and five consequent terms in extended belief rules are defined as follows:

$$U = \{0, 1, 2\}, \qquad D = \{-2, -1, 0, 1, 2\}$$

The initial EBRB is constructed using 5 numbers selected from the range of values for $x$ evenly, and is given in Table 3. The output of the initial EBRB system is shown in Fig. 4. In order to conduct EBRB training, a set of 100 data chosen evenly from the range of values for $x$ is used.

After training, a trained EBRB is obtained, which is shown in Table 4, the approximation curve and the actual function curve are shown in Fig. 5, and the change in the MSE is shown in Fig. 6.

As shown in Fig. 5 and Fig. 6, after training, the MSE of the EBRB system reduced from 0.0672 to 0.00036, which means the trained EBRB can more precisely and effectively provide an output for a given input. It should be noted that in Table 4, the trained rule weight of rule (3) is 0, which means it does not effect on the reasoning result of the EBRB system, thus, it is a redundant rule and could be deleted from the EBRB.

**Table 3**
Initial EBRB for Example 3.

| Attribute weight | Rule | Rule weight | $x$ | $f(x)$ | Antecedent attribute | Consequent |
|---|---|---|---|---|---|---|
| 1.0000 | 1 | 1.0000 | 0.0000 | 0.0000 | (1.0000, 0.0000, 0.0000) | (0.0000, 0.0000, 1.0000, 0.0000, 0.0000) |
| | 2 | 1.0000 | 0.5000 | 0.1237 | (0.5000, 0.5000, 0.0000) | (0.0000, 0.0000, 0.8763, 0.1237, 0.0000) |
| | 3 | 1.0000 | 1.0000 | 0.8415 | (0.0000, 1.0000, 0.0000) | (0.0000, 0.0000, 0.1585, 0.8415, 0.0000) |
| | 4 | 1.0000 | 1.5000 | 1.1671 | (0.0000, 0.5000, 0.5000) | (0.0000, 0.0000, 0.0000, 0.8329, 0.1671) |
| | 5 | 1.0000 | 2.0000 | -1.5136 | (0.0000, 0.0000, 1.0000) | (0.5136, 0.4864, 0.0000, 0.0000, 0.0000) |

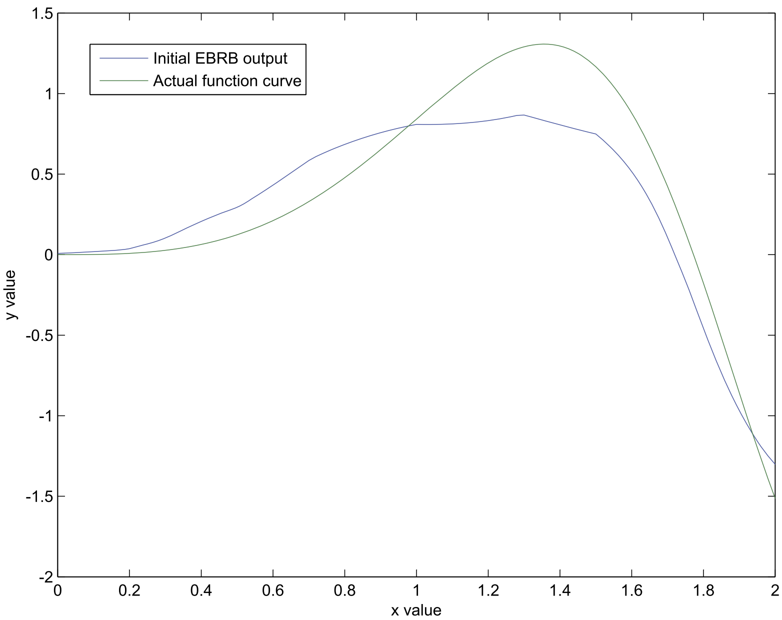


**Fig. 4.** Output of the initial EBRB system of Example 3.

**Table 4**
Trained EBRB of Example 3.

| Attribute weight | Rule | Rule weight | $x$ | $f(x)$ | Antecedent attribute | Consequent |
|---|---|---|---|---|---|---|
| 0.9164 | 1 | 1.0000 | 0.0000 | 0.0000 | (1.0000, 0.0000, 0.0000) | (0.0293, 0.0329, 0.8668, 0.0113, 0.0596) |
| | 2 | 0.9890 | 0.5000 | 0.2397 | (0.5039, 0.4922, 0.0039) | (0.1200, 0.0099, 0.7516, 0.1184, 0.0000) |
| | 3 | 0.0000 | 1.0000 | 0.8415 | (0.0028, 0.9944, 0.0028) | (0.1118, 0.0086, 0.0883, 0.6217, 0.1696) |
| | 4 | 0.9549 | 1.5000 | 1.4962 | (0.0000, 0.5000, 0.5000) | (0.0366, 0.0000, 0.0000, 0.0000, 0.9634) |
| | 5 | 0.8317 | 2.0000 | 1.8186 | (0.0000, 0.0000, 1.0000) | (0.9131, 0.0860, 0.0000, 0.0000, 0.0073) |

### *3.2. Analysis of EBRB structure*

As an expansion of the BRB system, EBRB system can work as a universal approximator that can approximate any real continuous function on a data set with acceptable accuracy. However, the number of extended belief rules in the EBRB could have a huge influence on the accuracy of the EBRB system. If the number of rules is lower than the threshold, i.e., the number of antecedent attribute referential grades $J$, the decrease of the number of rules would directly cause the decrease of the accuracy of the EBRB system. However, the increase of the number of rules in the EBRB may not necessarily lead to the increase of accuracy of the EBRB system, in some cases, excessive rules in the EBRB may lead to the increase of calculation time and even the decrease of the EBRB system accuracy. To illustrate the influence the number of rules in the EBRB has on its reasoning performance, the example below is studied.

**Example 4.** (EBRB reasoning performance under different number of rules): Suppose there is a single-input system shown in Fig. 7, the input is bounded in the interval [0, 1], and the output $y = x^2$ is a nonlinear increasing function of input $x$. To construct EBRB with different number of rules, the following information is provided.
(1) A set of eleven input-output data pairs are chosen as training data:

$$x_i = \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\} \tag{26}$$

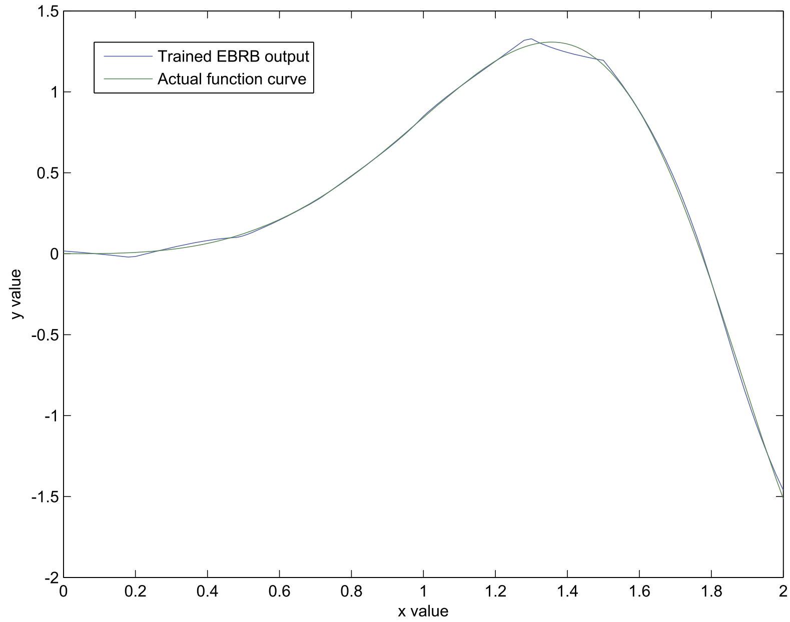


**Fig. 5.** Output of the trained EBRB system of Example 3.

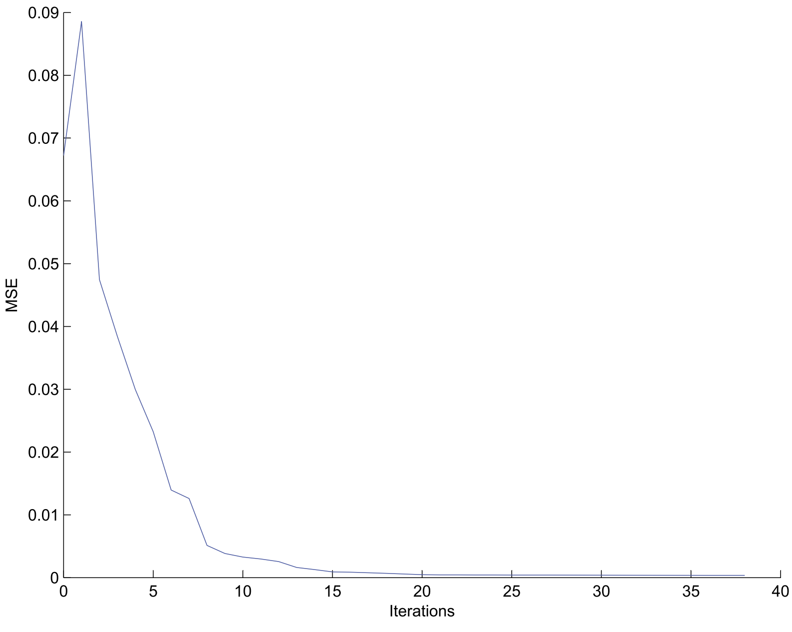


**Fig. 6.** Change in the MSE with the process of parameter training.

(2) The referential value of three antecedent grades as follows:

$$U = \{0, 0.5, 1\} \tag{27}$$

(3) The referential value of five consequent terms as follows:

$$D = \{-0.5, 0, 0.5, 1, 1.5\} \tag{28}$$

To illustrate the reasoning performance of EBRB system under different number of rules, three EBRB systems with 3, 6, and 11 rules are constructed, namely, EBRB-3, EBRB-6, and EBRB-11, respectively. Based on the EBRB training method in Section 3.1, all three EBRBs could be trained, and the trained parameters are listed in Table 5.

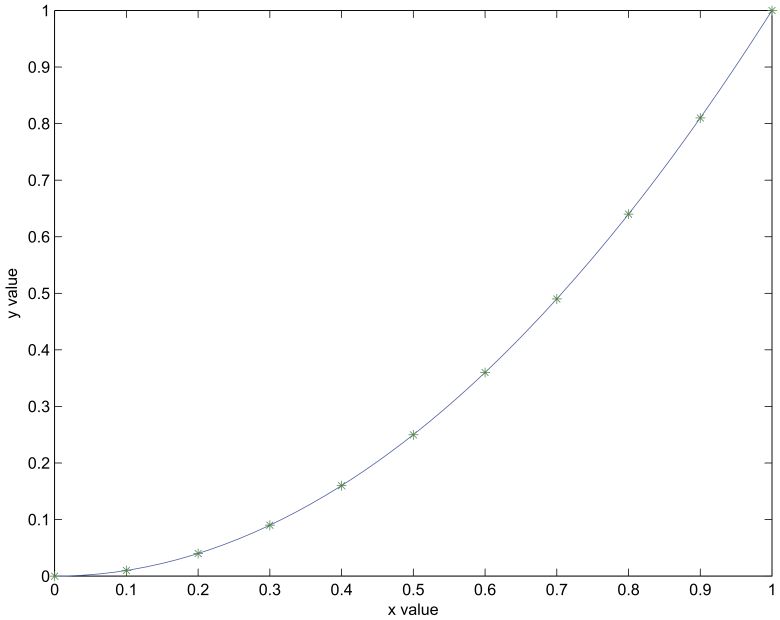


**Fig. 7.** Training data for Example 4.

**Table 5**
Three Trained EBRBs of Example 4.

| Type | Attribute weight | Rule No. | Rule weight | $x$ | $f(x)$ | Antecedent attribute | Consequent |
|---|---|---|---|---|---|---|---|
| EBRB-3 | 0.3851 | 1 | 1.0000 | 0.0000 | 0.0000 | (1.0000, 0.0000, 0.0000) | (0.2670, 0.6572, 0.0000, 0.0013, 0.0746) |
| | | 2 | 0.5006 | 0.5000 | 0.2500 | (0.2899, 0.4202, 0.2899) | (0.0206, 0.6274, 0.2597, 0.0000, 0.0923) |
| | | 3 | 0.4823 | 1.0000 | 1.0000 | (0.0000, 0.0000, 1.0000) | (0.0000, 0.0000, 0.0174, 0.6622, 0.3203) |
| EBRB-6 | 1.0000 | 1 | 1.0000 | 0.0000 | 0.0000 | (1.0000, 0.0000, 1.0000) | (0.0116, 0.9774, 0.0003, 0.0000, 0.0107) |
| | | 2 | 1.0000 | 0.2000 | 0.0400 | (0.6000, 0.4000, 0.0000) | (0.1992, 0.7217, 0.0341, 0.0114, 0.0336) |
| | | 3 | 0.4955 | 0.4000 | 0.1600 | (0.2484, 0.7031, 0.0484) | (0.3581, 0.6419, 0.0000, 0.0000, 0.0000) |
| | | 4 | 0.5293 | 0.6000 | 0.3600 | (0.0704, 0.6593, 0.2704) | (0.0249, 0.0038, 0.5043, 0.1723, 0.2946) |
| | | 5 | 0.9704 | 0.8000 | 0.6400 | (0.0938, 0.2124, 0.6938) | (0.2515, 0.0215, 0.3264, 0.1619, 0.2387) |
| | | 6 | 0.8344 | 1.0000 | 1.0000 | (0.0000, 0.0000, 1.0000) | (0.0000, 0.0000, 0.0000, 0.6127, 0.3873) |
| EBRB-11 | 1.0000 | 1 | 0.9992 | 0.0000 | 0.0000 | (1.0000, 0.0000, 0.0000) | (0.0582, 0.8921, 0.0000, 0.0009, 0.0488) |
| | | 2 | 1.0000 | 0.1000 | 0.0010 | (0.8000, 0.2000, 0.0000) | (0.0000, 0.9059, 0.0000, 0.0309, 0.0632) |
| | | 3 | 0.9997 | 0.2000 | 0.0040 | (0.6000, 0.4000, 0.0000) | (0.2459, 0.7464, 0.0077, 0.0000, 0.0000) |
| | | 4 | 1.0000 | 0.3000 | 0.0090 | (0.5577, 0.2846, 0.1577) | (0.3722, 0.6278, 0.0000, 0.0000, 0.0000) |
| | | 5 | 0.7575 | 0.4000 | 0.1600 | (0.2060, 0.7880, 0.0060) | (0.3703, 0.6297, 0.0000, 0.0000, 0.0000) |
| | | 6 | 0.5801 | 0.5000 | 0.2500 | (0.0000, 1.0000, 0.0000) | (0.0776, 0.5003, 0.3487, 0.0336, 0.0339) |
| | | 7 | 0.6536 | 0.6000 | 0.3600 | (0.1234, 0.5532, 0.3234) | (0.1147, 0.0407, 0.6240, 0.0906, 0.1300) |
| | | 8 | 0.9317 | 0.7000 | 0.4900 | (0.0025, 0.5950, 0.4025) | (0.0033, 0.0000, 0.6608, 0.0690, 0.2668) |
| | | 9 | 0.9060 | 0.8000 | 0.6400 | (0.0017, 0.3966, 0.6017) | (0.1092, 0.0097, 0.3850, 0.2584, 0.2377) |
| | | 10 | 0.9092 | 0.9000 | 0.8100 | (0.0008, 0.1983, 0.8009) | (0.2174, 0.0580, 0.0715, 0.5006, 0.1526) |
| | | 11 | 0.9972 | 1.0000 | 1.0000 | (0.0000, 0.0000, 1.0000) | (0.0076, 0.0033, 0.0009, 0.4459, 0.5423) |

**Table 6**
Comparison of MSE for three EBRBs of Example 4.

| Type | Number of rules | Training data set | Testing data set |
|---|---|---|---|
| EBRB-3 | 3 | 0.0019 | 0.0216 |
| EBRB-6 | 6 | 0.0000 | 0.0002 |
| EBRB-11 | 11 | 0.0001 | 0.0008 |

In order to compare the reasoning performance of these three EBRB systems, a set of 100 data generated from the interval $[0, 1]$ is used as the testing data. Table 6 shows the comparison of mean square error (MSE) of the three EBRB systems, and Fig. 8 shows the comparison between the results of these three EBRB systems. As shown in Table 6, when the number of rules increases from 3 to 6, the MSE of the training data decreases significantly from 0.0019 to 0.0000, and the MSE of the testing data also decreases from 0.0216 to 0.0002. However, when the number of rules increases from 6 to 11, the MSE of the training data and the testing data both increase, from 0.0000 to 0.0002 and from 0.0001 to 0.0008,

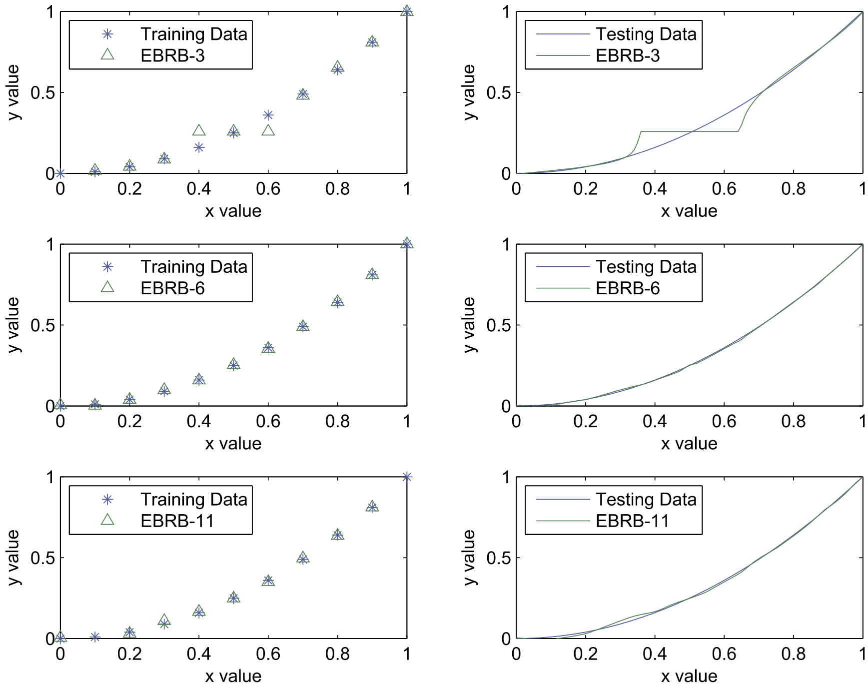


**Fig. 8.** Approximation curves of three EBRBs on training and testing data.

respectively, which means the accuracy of EBRB-6 is higher than that of EBRB-11. Fig. 8 clearly shows that too few rules will have a direct impact on the reasoning performance of the EBRB system, however, the increase of the number of rules does not necessarily lead to the increase of reasoning performance. In fact, as the number of rules increases from 6 to 11, it is clear in Fig. 8 that the reasoning performance of the EBRB system has been weakened. In this case, though EBRB-11 has more rules compared with EBRB-6, these rules are excessive information as they play a negative effect in determining the reasoning results of the EBRB system, thus, the five excess rules in EBRB-11 can be regarded as redundant rules and should be removed from the EBRB.

The numerical analysis of this section shows the reasoning performance of the EBRB system under different number of rules, it can be concluded that an EBRB with more rules may not lead to a better result, on the contrary, sometimes, it could lead to a worse result as excessive information may have negative influences on the EBRB system. Therefore, EBRB reduction method, which can search and fuse similar belief rules in the EBRB, should be studied to reduce the number of rules as well as eliminate excessive information.

### *3.3. EBRB reduction method based on DBSCAN algorithm*

As shown in Example 3, when the rule weight of a rule in EBRB is 0, it means that the rule won't play any role in determining the output of the EBRB system and thus can be regarded as redundant. Moreover, as shown in Example 4, when there are too many rules in the EBRB, excessive information could have a negative influence on its reasoning performance, and it will also cost excessive time and calculation resource to calculate the activation weight of each rule and obtain the final result. In many cases, there are belief rules that represent similar knowledge in the EBRB and can be considered redundant rules. Therefore, it is necessary to search and fuse similar belief rules in the EBRB to reduce its scale and improve its efficiency.

In the aforementioned case, only a few extended belief rules are single rules, while most extended belief rules are similar to each other. Using the DBSCAN algorithm proposed by Ester et al. [52], similar rules can be determined by calculating their distance, and the EBRB reduction method based on the DBSCAN algorithm can be proposed. The process of proposed EBRB reduction method is summarized as Algorithm 1.

It should be noted that the Euclidean distance $d_{k,m}$ between the rule $R_k$ and $R_m$ is used to measure the distance of each antecedent attribute between these two rules, which could be calculated using Eq. (8). It's also worth noting that in this process, the parameter $\boldsymbol{\varepsilon}$ used as the threshold to determine whether the two belief rules should be fused is the *eps* in DBSCAN algorithm, which is the main parameter determining the result of the reduction process. In order to effectively fuse redundant belief rules without the cost of losing information, $\boldsymbol{\varepsilon}$ should be determined based on the actual situation of the EBRB system before the reduction process.

**Algorithm 1** EBRB reduction based on DBSCAN algorithm.

**Input:** The initial EBRB ($\Omega$), unvisited queues ($Q_{uv}$), visiting queues ($Q_v$), result queues ($Q_r$), fusion threshold ($\boldsymbol{\varepsilon} = \{\varepsilon_1, \ldots, \varepsilon_J\}$).
**Output:** The new EBRB ($\Omega'$).
$Q_{uv} = \Omega$, $\Omega' = \phi$
**for** $R_t \in Q_v$ **do**
  $Q_{uv} = Q_{uv} - \{R_k\}$, $Q_v = \{R_k\}$, $Q_r = \phi$
  **for** $R_k \in Q_v$ **do**
    **for** $R_m \in Q_{uv}$ **do**
      Calculate absolute error $d_{k,m} =< d^1_{k,m}, d^2_{k,m}, \ldots, d^J_{k,m} >$, where $d^j_{k,m} = \sqrt{\sum_{i=1}^{N} (\alpha^k_{ij} - \alpha^m_{ij})^2}$
      **for** every $d^j_{k,m}$ in the absolute error $d_{k,m}$ **do**
        **if** $d^j_{k,m} < \varepsilon_j$ **then**
          $Q_{uv} = Q_{uv} - \{R_m\}$, $Q_v = Q_v + \{R_m\}$
        **end if**
      **end for**
    **end for**
    $Q_v = Q_v - \{R_k\}$, $Q_r = Q_r + \{R_k\}$
  **end for**
  Generate new rule $R_t{}'$ using rules from $Q_r$
  $\Omega' = \Omega' + \{R_t{}'\}$
**end for**

**Remark 5.** When belief structures $S_1 = \{(A_{1,j}, \alpha_{1,j}), j = 1, \ldots, J\}$, $S_2 = \{(A_{2,j}, \alpha_{2,j}), j = 1, \ldots, J\}$ satisfy $|\alpha_{1,j} - \alpha_{2,j}| < 0.5$, $j = 1 \ldots, J$, i.e. $\hat{d} < 0.7$, they are regarded as *close*. In general, the fusion threshold $\boldsymbol{\varepsilon}$ is set to $\boldsymbol{\varepsilon} < 0.5\hat{d} = 0.35$. Normally, for problems with less than three antecedent attributes, to avoid loss of information caused by too many rules being fused, $\boldsymbol{\varepsilon}$ is set to $\boldsymbol{\varepsilon} < 0.1$.

After the belief rules in $Q_r$ is obtained, a new belief rule and its parameters, including rule weight, antecedent attribute belief degrees and consequent belief degrees, will be determined by fusing the rules in $Q_r$. Suppose there are $L$ rules in $Q_r$, and the procedure can be represented as follows:

(1) The belief degree of the $i$th referential value of the $j$th antecedent attribute of the new rule can be obtained by calculating the average of the $L$ rules, and is represented as follows:

$$\alpha_{ij} = \frac{\sum_{k=1}^{L} \alpha^k_{ij}}{L} \tag{29}$$

(2) The rule weight of the new rule can be obtained using all $L$ belief rules as follows:

$$\theta = \frac{\sum_{k=1}^{L} \theta_k}{L} \tag{30}$$

(3) The belief degrees of the new rule can be obtained using Eq. (12). It should be noted that in this case, $w_k$ is not the activation weight, instead, it is the relative rule weight of the $k$th rule, and it can be calculated as follows:

$$w_k = \frac{\theta_k}{\sum_{k=1}^{L} \theta_k} \tag{31}$$

Following this procedure, all the redundant rules in the EBRB can be effectively fused and a new EBRB could be obtained. However, though the fused rules are similar in distance, there may still be small differences in the information they express, thus, simply using the reduced EBRB system may not always reach a precise result. Therefore, it is necessary to conduct EBRB training for the reduced EBRB to enhance its performance.

### 3.4. EBRB reduction and training process

With the EBRB training method and EBRB reduction method being defined respectively, the EBRB reduction and training process is shown in Fig. 9, and be concluded in three steps as follows:

**Step 1:** Initial EBRB generation

In order to construct an EBRB system, the initial EBRB needs to be constructed using the EBRB generation method proposed by Liu et al. [36], and a set of input-output data pairs, either from system observations or based on experts knowledge, are used as the training set in the EBRB generation process. Thus, an initial EBRB with excessive belief rules can be obtained.

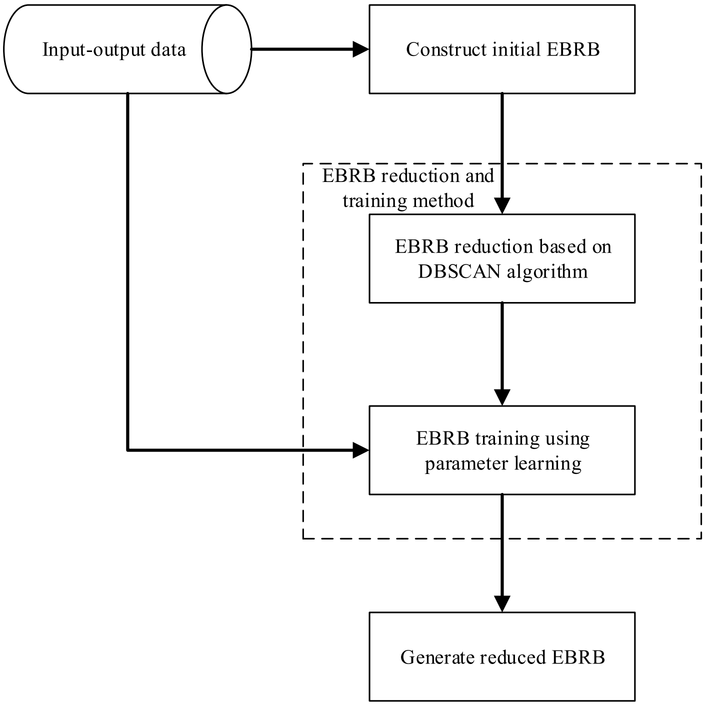


**Fig. 9.** Process of the EBRB reduction and training method.

**Step 2:** Rule reduction for the EBRB

After the initial EBRB is generated using a set of training data, there could be belief rules that are similar enough to be regarded as redundant rules expressing the same information. Thus, using the EBRB reduction method proposed in Section 3.3, redundant rules in the initial EBRB would be searched and fused, which results in the size of the EBRB being reduced.

**Step 3:** Parameter learning for the reduced EBRB

After EBRB reduction, the newly obtained EBRB should be trained before being used to conduct reasoning process. Therefore, the parameter learning method introduced in Section 3.1 could be applied to the reduced EBRB to enhance its performance without the loss of information.

As aforementioned, after the EBRB reduction and training process, the size of the initial large-scale EBRB systems can be reduced as its redundant rules removed from the EBRB, while its accuracy maintains the same level, in many cases, could even be improved.

## 4. Case studies

In order to illustrate the performance and effectiveness of the proposed EBRB reduction and training method, two cases of regression and classification problems are studied.

### 4.1. Rule reduction for regression problem

#### 4.1.1. Function approximation

Consider the following nonlinear function:

$$y = x * \sin(x^2) + x * \sin(x) \tag{32}$$

where $x$ is bounded in the interval $[0, \pi]$.

This is a typical single input-output system, with $x$ being the input and $y$ being the output. Suppose $x$ being the antecedent attribute and $y$ being the consequent, an EBRB can be used to predict the value of $y$ based on the value $x$.

Six antecedent grades are used to define $x$, namely, Extremely Small (ES), Very Small (VS), Small (S), Large (L), Very Large (VL), and Extremely Large (EL). The corresponding values are as follows:

$$\begin{aligned} U &\in \{ES, VS, S, L, VL, EL\} \\ &= \{0, 0.7, 1.4, 2.1, 2.8, 3.5\} \end{aligned} \tag{33}$$

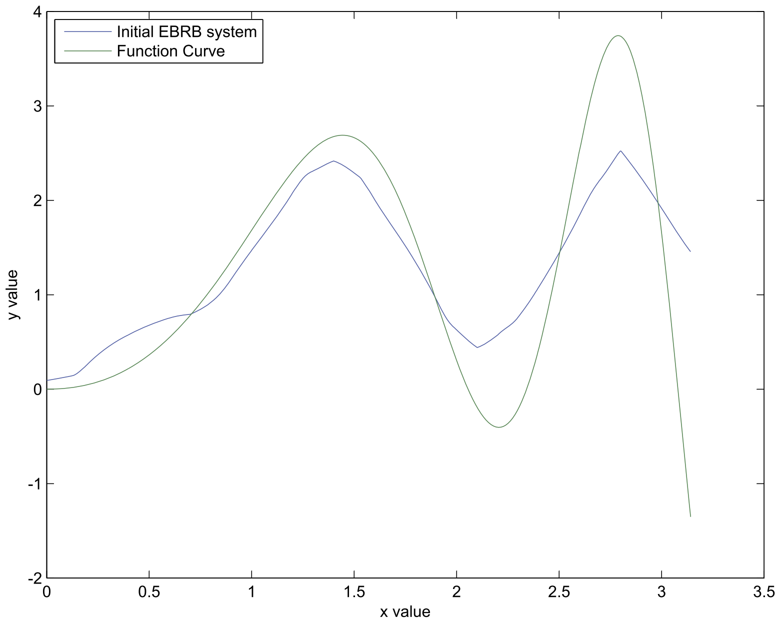


**Fig. 10.** Initial EBRB system output.

Similarly, six consequent grades are used to define $y$, namely, Negative Medium (NM), Negative Small (NS), Positive Tiny (PT), Positive Small (PS), Positive Medium (PM), and Positive Large (PL), and the corresponding values are expressed as follows:

$$\begin{aligned} D &\in \{NM, NS, PT, PS, PM, PL\} \\ &= \{-2, -0.8, 0.4, 1.6, 2.8, 4\} \end{aligned} \tag{34}$$

**(1) EBRB generation**

The first step of the EBRB reduction and training method is to generate the initial EBRB using the training data set. To generate the initial EBRB, 200 numbers from the range of values for $x$ are chosen randomly as the training data, and the referential values of $x$ and $y$ are shown above. To test the output of the EBRB system, 1000 numbers chosen from the range of $x$ evenly are used as the testing data, and the output of the initial EBRB system is shown in Fig. 10.

**(2) EBRB reduction**

After the initial EBRB is established, EBRB reduction method can be applied to the initial EBRB to search and fuse similar belief rules following the process in Section 3.3. In this case, the fusion threshold $\varepsilon$ is set to 0.05, so similar belief rules within the distance of $\varepsilon$ would be fused. After EBRB reduction, the number of rules in the EBRB reduces from 200 to 40, a significant decrease of 80%. However, after reduction, the reasoning accuracy of the EBRB system decreases, with the MSE of training data slightly increasing from 0.0961 to 0.1024, which is caused by the increase of uncertainty of reduction process.

**(3) EBRB training**

Once the reduced EBRB is obtained, parameter learning method is applied to conduct EBRB training and enhance its reasoning performance. In this case, the same data set used to generate the initial EBRB is used as the training data set, i.e. the 200 input-output pairs chosen randomly from the range of values for $x$. As shown in Fig. 11, after 88 iterations of training, the MSE decreases from 0.1024 to 0.0018, and the function regression result is shown in Fig. 12.

*4.1.2. Comparative analysis*

In order to illustrate the efficiency of the proposed method in function approximation problems, Liu-EBRB [36], VP-EBRB and MVP-EBRB [44] are also used for the same function approximation problem, and the results of Liu-EBRB, VP-EBRB, MVP-EBRB, and the proposed DA-EBRB are shown in Fig. 12. From Fig. 12, it can be seen that there is a significant difference between the output of Liu-EBRB system and the actual function output. Though the performance of VP-EBRB system and MVP-EBRB system are significantly better compared with Liu-EBRB, the difference between these systems and actual function output are still larger than that of DA-EBRB system. Thus, it can be concluded that the proposed DA-EBRB can more precisely approximate nonlinear function than Liu-EBRB, VP-EBRB, and MVP-EBRB.

Using 1000 testing data, the performance of DA-EBRB, Liu-EBRB, VP-EBRB and MVP-EBRB are further compared in following aspects: (1) MSE: the mean squared error between the system output and actual output; (2) MAE: the mean absolute error between the system output and actual output; (3) Visited Rules: the number of rules visited in the inference process, and the results are summarized in Table 7. As shown in Table 7, DA-EBRB is significantly better than Liu-EBRB, VP-EBRB, and

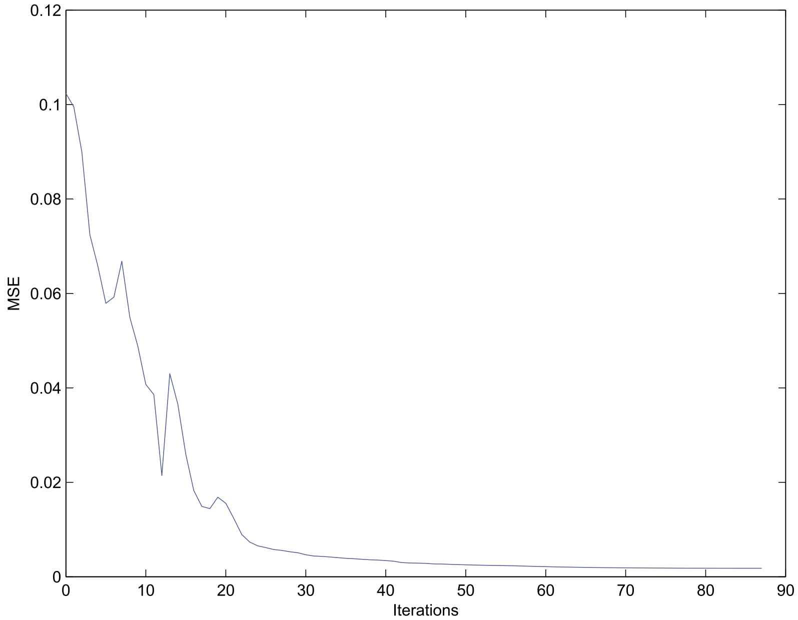


**Fig. 11.** Change in the MSE in the process of EBRB training.

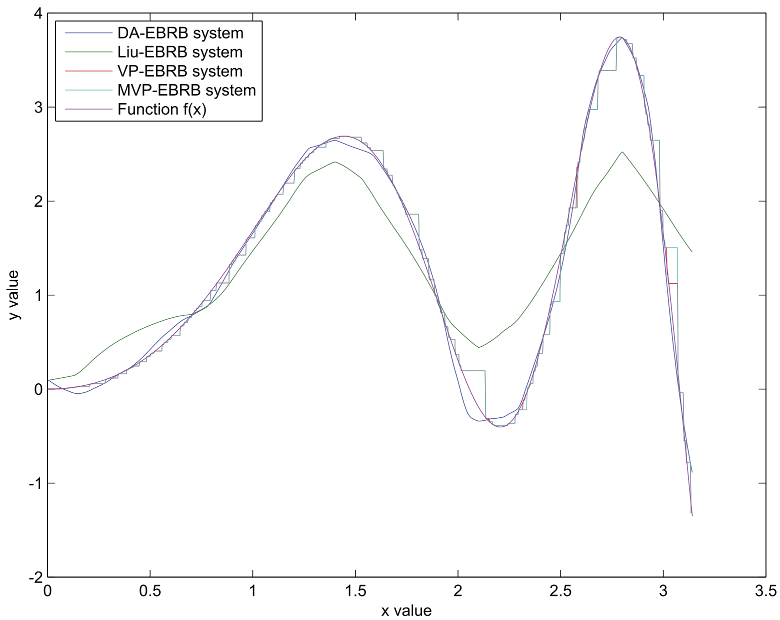


**Fig. 12.** Approximate curve of different EBRB systems.

**Table 7**
Comparison with other EBRBs.

| | MSE | MAE | Visited rules |
|---|---|---|---|
| DA-EBRB | 0.0087 | 0.0687 | 40000 |
| Liu-EBRB | 0.4032 | 0.4667 | 200000 |
| VP-EBRB | 0.0157 | 0.0627 | 76052 |
| MVP-EBRB | 0.0198 | 0.0749 | 40898 |

**Table 8**
Details of the classification datasets.

| No. | Name | # Instances | # Attributes | # Classes | # before reduction | # after reduction |
|---|---|---|---|---|---|---|
| 1 | Iris | 150 | 4 | 3 | 135 | 31.6 |
| 2 | Transfusion | 748 | 4 | 2 | 673.2 | 20 |
| 3 | Seeds | 210 | 7 | 3 | 189 | 72.5 |
| 4 | Ecoli | 336 | 7 | 8 | 302.4 | 94.6 |
| 5 | Yeast | 1484 | 8 | 10 | 1335.6 | 173.5 |
| 6 | Glass | 214 | 9 | 6 | 237.8 | 99.9 |
| 7 | Diabetes | 768 | 8 | 2 | 691.2 | 290.7 |

MVP-EBRB. Taking the MSE for example, the MSE of DA-EBRB is 0.0087, while the MSE of Liu-EBRB, VP-EBRB, and MVP-EBRB are 0.4032, 0.0198, and 0.0157, respectively, which is about half of the lowest MSE of Liu-EBRB, VP-EBRB, and MVP-EBRB. Furthermore, DA-EBRB visits fewer rules during the inference process than that of Liu-EBRB, VP-EBRB, and MVP-EBRB, which could effectively reduce the time cost of the inference process.

In summary, the comparison result from the function approximation problem proves that the proposed EBRB reduction and training method can reduce the size of EBRB and improve its accuracy at the same time, thus, the proposed DA-EBRB is more accurate and efficient.

## 4.2. *Classification problem*

### 4.2.1. *Data sets and experiment conditions*

In this case, all datasets used are obtained from UCI machine-learning repository [53], which has been widely used in testing the performance of different kinds of classifiers. The details of the classification datasets used in this case are shown in Table 8, which also shows the number of belief rules in the EBRB before and after the EBRB reduction.

To construct the EBRB, suppose that each antecedent attribute has three antecedent grades and the consequent has the same number of grades as the number of classes. In addition, each dataset was randomized by permuting its samples in order to perform a series of 10-fold cross-validation tests for DA-EBRB system.

### 4.2.2. *Comparative analysis with the conventional EBRB systems*

In order to illustrate the effectiveness of the proposed method, conventional EBRB system and its improvements such as the improved EBRB by the DRA method [43], the VP method, the MVP method and the CABRA method [45], as well as the improved EBRB classification system proposed by Yang et al. [41], are used to compare with the proposed DA-EBRB system, where all these EBRB systems are abbreviated as Liu-EBRB, DRA-EBRB, VP-EBRB, MVP-EBRB, CABRA-EBRB and EBRB-C, respectively. Moreover, in order to demonstrate the result of the parameter training process, the classification result of the reduced EBRB system, denoted as EBRB-R, is also illustrated. The results of each dataset and method are measured with: (1) Accuracy: percentage of classes correctly classified from the total simples; (2) VRR: percentage of visited rules in the inference process from the total rules generated from the training set; (3) Failed: number of tests where the method could not activate any rules. The comparison results are shown in Table 9.

As Table 9 illustrates, compared with existing EBRB systems, the accuracy of DA-EBRB system is better than that of the other EBRB systems for most classification datasets, and it visits fewer belief rules during inference process, with no failed data. For example, for Glass dataset, the accuracy of DA-EBRB is 77.57%, which is significantly better than Liu-EBRB (67.85%), DRA-EBRB (69.65%), VP-EBRB (71.75%), MVP-EBRB (72.06%), CABRA-EBRB (72.90%) and EBRB-C (75.51%), while the VRR is 53.75%, which is also better than that of Liu-EBRB, DRA-EBRB, VP-EBRB, MVP-EBRB, CABRA-EBRB, and EBRB-C. Moreover, comparison between the results of DA-EBRB and EBRB-R shows that the EBRB training process is significant in improving its accuracy, take Glass dataset for example, the accuracy of EBRB-R is 57.07%, while DA-EBRB reaches a 77.57% accuracy, which is a significant 35.92% increase. It's also worth noting that the performance of DA-EBRB is especially good when the dataset is relatively large, for instance, the VRR of DA-EBRB of Transfusion is significantly lower than other methods, with the VRR being 3.39%, while also maintaining relatively good accuracy.

### 4.2.3. *Comparative analysis with conventional methods*

To further verify the validity of the proposed EBRB reduction and training method, the classification results derived from the 10-fold cross-validation are further compared with conventional machine learning methods, as shown in Table 10.

By comparing the results of conventional methods and the proposed EBRB reduction and training method, it is proved that the proposed method can provide satisfactory results. For example, for Glass, Ecoli, and Diabetes datasets, the proposed method can obtain 86.01%, 77.57% and 83.67% accuracy, which outperforms any other methods. For Iris dataset, though it may seem to be less satisfactory, it still reaches the third best accuracy, and its accuracy is not significant lower.

Furthermore, some implicit information can also be found from the comparison:

(1) The proposed DA-EBRB system has superior accuracy in handling multi-class datasets, as it is clear from Table 10 that the accuracy of Glass and Ecoli datasets are significantly better than other methods.

**Table 9**
Comparison of the results for classification datasets.

| | Aspects | Iris | Transfusion | Seeds | Ecoli | Yeast | Glass | Diabetes |
|---|---|---|---|---|---|---|---|---|
| Liu-EBRB | Accuracy | 95.33 | 76.14 | 91.33 | 81.16 | 45.61 | 67.85 | 73.39 |
| | VRR | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| | Failed | 0 | 0 | 0 | 1 | 0 | 3.35 | 0 |
| DRA-EBRB | Accuracy | 95.50 | 76.57 | 92.02 | 83.76 | 54.13 | 69.65 | 71.44 |
| | VRR | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| | Failed | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| VP-EBRB | Accuracy | 95.13 | 77.33 | 92.57 | 84.87 | 58.15 | 71.75 | 71.89 |
| | VRR | 39.43 | 16.69 | 33.76 | 44.69 | 40.43 | 57.13 | 26.04 |
| | Failed | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| MVP-EBRB | Accuracy | 95.87 | **80.36** | 92.38 | 85.61 | 57.49 | 72.06 | 72.59 |
| | VRR | 78.60 | 42.63 | 81.88 | 51.18 | 41.76 | 84.54 | 69.52 |
| | Failed | 0 | 0 | 0 | 1 | 0 | 2 | 0 |
| CABRA-EBRB | Accuracy | 96.00 | 72.07 | 92.38 | 85.24 | 52.70 | 72.90 | 76.34 |
| | VRR | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| | Failed | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| EBRB-C | Accuracy | 95.73 | 76.88 | 93.24 | **87.14** | 54.51 | 75.51 | 75.98 |
| | VRR | 100 | 100 | 100 | 100 | 100 | 100 | 100 |
| | Failed | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| DA-EBRB | Accuracy | **97.33** | 77.81 | **95.62** | 86.01 | **59.10** | **77.57** | **83.67** |
| | VRR | 23.40 | 3.39 | 45.61 | 33.65 | 14.32 | 53.75 | 45.30 |
| | Failed | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| EBRB-R | Accuracy | 92.33 | 57.33 | 95.00 | 81.24 | 52.65 | 57.07 | 76.22 |
| | VRR | 23.40 | 3.39 | 45.61 | 33.65 | 14.32 | 53.75 | 45.30 |
| | Failed | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 10**
Comparison of accuracy of conventional methods.

| Methods | Core supporting theory | Iris | Glass | Ecoli | Diabetes |
|---|---|---|---|---|---|
| kNN [54] | K nearest neighbor | 85.17 | 66.87 | 81.27 | 73.06 |
| SpectralCAT [55] | Categorical spectral clustering | 97.00 | 70.00 | 74.00 | |
| AISWNB [56] | Naive Bayes | 94.87 | 57.74 | | 75.86 |
| FRBCS [57] | Fuzzy rule base | 92.67 | 66.05 | 76.12 | 67.95 |
| WLTSVM [58] | Support vector machine | **98.00** | 49.19 | | |
| NBSVM [59] | Support vector machine | 96.00 | 66.07 | | 77.77 |
| LMT [60] | Logistic model tree | 96.20 | 69.70 | | 77.10 |
| HHONC [61] | Netural network | 97.46 | 56.50 | | 79.15 |
| DA-EBRB | Belief rule base | 97.33 | **77.57** | **86.01** | **83.67** |

(2) There is no universal method that can reach the best accuracy for all methods, as many factors such as internal structure and noise data are likely to affect the classification accuracy.

## 5. Conclusion

A new EBRB reduction and training method is proposed in this paper to solve the problem that too many belief rules of the oversize EBRB could increase the calculation time cost and affect the reasoning performance. The main conclusions of this paper can be summarized as follows:

(1) The reasoning performance of EBRB systems with different number of belief rules under the same number of training data are examined. This paper reveals the influence the number of belief rules has on the reasoning performance of the EBRB system, and concludes that oversize EBRB could have a negative influence on the reasoning performance of the EBRB system.
(2) Based on the DBSCAN algorithm, the EBRB reduction method is proposed. In the proposed method, redundant rules are searched and fused to reduce the number of rules in the oversize EBRB. Using this method, similar belief rules within the distance of the fusion threshold could be fused and thus effectively reduce the size of EBRB.
(3) The previous studies simply used training data to generate EBRB and removed the EBRB training process, which significantly lowered the accuracy of EBRB system. Thus, this paper proposed the EBRB training method using parameter learning, which uses the same training data to train parameters of the EBRB system, including antecedent attribute belief degrees, consequent belief degrees, attribute weights and rule weights, and is shown to be able to improve the accuracy of EBRB system.

In addition, two case studies in regression and classification are used to illustrate the validation of the proposed method, and results show that the proposed method can effectively reduce the size of EBRB and improve its reasoning performance. In summary, the experiments demonstrate the feasibility and effectiveness of the proposed method.

For future research, we will investigate how to improve the reasoning performance of incomplete EBRB, and we believe rule adding method is an efficient way to guarantee the completeness and effectiveness of incomplete EBRB. Furthermore, in the EBRB reduction process, the fusion threshold is important to the result, and how to appropriately select the fusion threshold would also be further studied.

Moreover, as shown in this paper, the design of the original EBRB systems could be further studied as the similarity measure and distance measure are not strictly defined. Though several modifications have been proposed, further researches on the modification of the original EBRB systems would be considered.

## Declaration of competing interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

## Acknowledgement

The authors thanks the two anonymous reviewers for their useful comments. This work was supported by the National Natural Science Foundation of China (Nos. 61573283, 61903305), the Key Laboratory Open Foundation of Data Link Technology (No. CLDL-20182113) and the Research Funds for Interdisciplinary Subject, NWPU.

## References

[1] R. Sun, Robust reasoning: integrating rule-based and similarity-based reasoning, Artif. Intell. 75 (1995) 241–295.
[2] M. Negnevitsky, Artificial Intelligence: A Guide to Intelligent Systems, Pearson Education, 2005.
[3] A. Ligeza, Logical Foundations for Rule-Based Systems, Vol. 11, Springer, 2006.
[4] S.M. Sulzberger, N. Tschichold-Gurman, S.J. Vestli, Fun: optimization of fuzzy rule based systems using neural networks, in: IEEE International Conference on Neural Networks, 1993, IEEE, 1993, pp. 312–316.
[5] P.A. Castro, H.A. Camargo, Learning and optimization of fuzzy rule base by means of self-adaptive genetic algorithm, in: 2004 IEEE International Conference on Fuzzy Systems, 2004. Proceedings, vol. 2, IEEE, 2004, pp. 1037–1042.
[6] S.-J. Kang, C.-H. Woo, H.-S. Hwang, K.B. Woo, Evolutionary design of fuzzy rule base for nonlinear system modeling and control, IEEE Trans. Fuzzy Syst. 8 (2000) 37–45.
[7] M. Setnes, R. Babuska, U. Kaymak, H.R. van Nauta Lemke, Similarity measures in fuzzy rule base simplification, IEEE Trans. Syst. Man Cybern., Part B, Cybern. 28 (1998) 376–386.
[8] Y. Yam, P. Baranyi, C.-T. Yang, Reduction of fuzzy rule base via singular value decomposition, IEEE Trans. Fuzzy Syst. 7 (1999) 120–132.
[9] M. Setnes, R. Babuska, Rule base reduction: some comments on the use of orthogonal transforms, IEEE Trans. Syst. Man Cybern., Part C, Appl. Rev. 31 (2001) 199–206.
[10] J.-B. Yang, J. Liu, J. Wang, H.-S. Sii, H.-W. Wang, Belief rule-base inference methodology using the evidential reasoning approach-RIMER, IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum. 36 (2006) 266–285.
[11] A. Dempster, A generalization of bayesian inference, J. R. Stat. Soc., Ser. B, Methodol. 30 (1968) 205–247.
[12] G. Shafer, A Mathematical Theory of Evidence, Vol. 42, Princeton university press, 1976.
[13] T. Denoeux, Decision-making with belief functions: a review, Int. J. Approx. Reason. 109 (2019) 87–110.
[14] J.M. Bernardo, A.F. Smith, Bayesian Theory, vol. 405, John Wiley & Sons, 2009.
[15] L.A. Zadeh, et al., Fuzzy sets, Inf. Control 8 (1965) 338–353.
[16] G. Klir, B. Yuan, Fuzzy Sets and Fuzzy Logic, vol. 4, Prentice Hall, New Jersey, 1995.
[17] R.C. Núñez, M.N. Murthi, K. Premaratne, M. Scheutz, O. Bueno, Uncertain logic processing: logic-based inference and reasoning using Dumpster–Shafer models, Int. J. Approx. Reason. 95 (2018) 1–21.
[18] Y.-M. Wang, J.-B. Yang, D.-L. Xu, K.-S. Chin, Consumer preference prediction by using a hybrid evidential reasoning and belief rule-based methodology, Expert Syst. Appl. 36 (2009) 8421–8430.
[19] Y. Yang, C. Fu, Y.-W. Chen, D.-L. Xu, S.-L. Yang, A belief rule based expert system for predicting consumer preference in new product development, Knowl.-Based Syst. 94 (2016) 105–113.
[20] J.-B. Yang, Y.-M. Wang, D.-L. Xu, K.-S. Chin, L. Chatton, Belief rule-based methodology for mapping consumer preferences and setting product targets, Expert Syst. Appl. 39 (2012) 4749–4759.
[21] G. Kong, D.-L. Xu, X. Liu, J.-B. Yang, Applying a belief rule-base inference methodology to a guideline-based clinical decision support system, Expert Syst. 26 (2009) 391–408.
[22] Z.-G. Zhou, F. Liu, L.-L. Li, L.-C. Jiao, Z.-J. Zhou, J.-B. Yang, Z.-L. Wang, A cooperative belief rule based decision support system for lymph node metastasis diagnosis in gastric cancer, Knowl.-Based Syst. 85 (2015) 62–70.
[23] S. Rahaman, Diabetes diagnosis expert system by using belief rule base with evidential reasoning, in: 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), IEEE, 2015, pp. 1–6.
[24] G. Li, Z. Zhou, C. Hu, L. Chang, Z. Zhou, F. Zhao, A new safety assessment model for complex system based on the conditional generalized minimum variance and the belief rule base, Saf. Sci. 93 (2017) 108–120.
[25] C. Sun, C. Wu, D. Chu, Z. Lu, J. Tan, J. Wang, A recognition model of driving risk based on belief rule-base methodology, Int. J. Pattern Recognit. Artif. Intell. 32 (2018) 1850037.
[26] J. Zhang, X. Yan, D. Zhang, S. Haugen, X. Yang, Safety management performance assessment for maritime safety administration (MSA) by using generalized belief rule base methodology, Saf. Sci. 63 (2014) 157–167.
[27] G. Li, Z. Zhou, C. Hu, L. Chang, H. Zhang, C. Yu, An optimal safety assessment model for complex systems considering correlation and redundancy, Int. J. Approx. Reason. 104 (2019) 38–56.
[28] Z.-J. Zhou, C.-H. Hu, J.-B. Yang, D.-L. Xu, D.-H. Zhou, Online updating belief-rule-base using the rimer approach, IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum. 41 (2011) 1225–1243.

[29] J.-B. Yang, J. Liu, D.-L. Xu, J. Wang, H. Wang, Optimization models for training belief-rule-based systems, IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum. 37 (2007) 569–585.
[30] D.-L. Xu, J. Liu, J.-B. Yang, G.-P. Liu, J. Wang, I. Jenkinson, J. Ren, Inference and learning methodology of belief-rule-based expert system for pipeline leak detection, Expert Syst. Appl. 32 (2007) 103–113.
[31] L. Chang, Z. Zhou, Y. You, L. Yang, Z. Zhou, Belief rule based expert system for classification problems with new rule activation and weight calculation procedures, Inf. Sci. 336 (2016) 75–91.
[32] L. Chang, Y. Zhou, J. Jiang, M. Li, X. Zhang, Structure learning for belief rule base expert system: a comparative study, Knowl.-Based Syst. 39 (2013) 159–172.
[33] X. Ke, L. Ma, Y. Wang, A modified belief rule based model for uncertain nonlinear systems identification, J. Intell. Fuzzy Syst. 32 (2017) 3879–3891.
[34] L. Chang, Y. Chen, Z. Hao, Z. Zhou, X. Xu, X. Tan, Indirect disjunctive belief rule base modeling using limited conjunctive rules: two possible means, Int. J. Approx. Reason. 108 (2019) 1–20.
[35] Y.-M. Wang, L.-H. Yang, Y.-G. Fu, L.-L. Chang, K.-S. Chin, Dynamic rule adjustment approach for optimizing belief rule-base expert system, Knowl.-Based Syst. 96 (2016) 40–60.
[36] J. Liu, L. Martinez, A. Calzada, H. Wang, A novel belief rule base representation, generation and its inference methodology, Knowl.-Based Syst. 53 (2013) 129–141.
[37] A. Calzada, J. Liu, H. Wang, C. Nugent, L. Martinez, Application of a spatial intelligent decision system on self-rated health status estimation, J. Med. Syst. 39 (2015) 138.
[38] M. Espinilla, J. Medina, A. Calzada, J. Liu, L. Martínez, C. Nugent, Optimizing the configuration of an heterogeneous architecture of sensors for activity recognition, using the extended belief rule-based inference methodology, Microprocess. Microsyst. 52 (2017) 381–390.
[39] L.-H. Yang, J. Liu, Y.-M. Wang, L. Martínez, Comparative analysis on extended belief rule-based system for activity recognition, in: Data Science and Knowledge Engineering for Sensing Decision Support: Proceedings of the 13th International FLINS Conference (FLINS 2018), vol. 11, World Scientific, 2018, p. 430.
[40] L.-H. Yang, J. Liu, Y.-M. Wang, L. Martínez, A micro-extended belief rule-based system for big data multiclass classification problems, IEEE Trans. Syst. Man Cybern. Syst. (2018), https://doi.org/10.1109/TSMC.2018.2872843.
[41] L.-H. Yang, J. Liu, Y.-M. Wang, L. Martínez, Extended belief-rule-based system with new activation rule determination and weight calculation for classification problems, Appl. Soft Comput. 72 (2018) 261–272.
[42] A. Calzada, J. Liu, H. Wang, A. Kashyap, Dynamic rule activation for extended belief rule bases, in: 2013 International Conference on Machine Learning and Cybernetics (ICMLC), vol. 4, IEEE, 2013, pp. 1836–1841.
[43] A. Calzada, J. Liu, H. Wang, A. Kashyap, A new dynamic rule activation method for extended belief rule-based systems, IEEE Trans. Knowl. Data Eng. 27 (2015) 880–894.
[44] Y.-Q. Lin, Y.-G. Fu, Q. Su, Y.-M. Wang, X.-T. Gong, A rule activation method for extended belief rule base with VP-tree and MVP-tree, J. Intell. Fuzzy Syst. 33 (2017) 3695–3705.
[45] L.-H. Yang, Y.-M. Wang, Y.-G. Fu, A consistency analysis-based rule activation method for extended belief-rule-based systems, Inf. Sci. 445 (2018) 50–65.
[46] L.-H. Yang, Y.-M. Wang, Y.-X. Lan, L. Chen, Y.-G. Fu, A data envelopment analysis (DEA)-based method for rule reduction in extended belief-rule-based systems, Knowl.-Based Syst. 123 (2017) 174–187.
[47] L.-H. Yang, J. Liu, Y.-M. Wang, L. Martínez, New activation weight calculation and parameter optimization for extended belief rule-based system based on sensitivity analysis, Knowl. Inf. Syst. 60 (2019) 837–878.
[48] J.-B. Yang, M.G. Singh, An evidential reasoning approach for multiple-attribute decision making with uncertainty, IEEE Trans. Syst. Man Cybern. 24 (1994) 1–18.
[49] J.-B. Yang, D.-L. Xu, On the evidential reasoning algorithm for multiple attribute decision analysis under uncertainty, IEEE Trans. Syst. Man Cybern., Part A, Syst. Hum. 32 (2002) 289–304.
[50] J.-B. Yang, Y.-M. Wang, D.-L. Xu, K.-S. Chin, The evidential reasoning approach for MADA under both probabilistic and fuzzy uncertainties, Eur. J. Oper. Res. 171 (2006) 309–343.
[51] A.-L. Jousselme, P. Maupin, Distances in evidence theory: comprehensive survey and generalizations, Int. J. Approx. Reason. 53 (2012) 118–145.
[52] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al., A density-based algorithm for discovering clusters in large spatial databases with noise, in: KDD 96, 1996, pp. 226–231.
[53] D. Dheeru, E. Karra Taniskidou, UCI machine learning repository, http://archive.ics.uci.edu/ml, 2017.
[54] J. Derrac, F. Chiclana, S. García, F. Herrera, Evolutionary fuzzy k-nearest neighbors algorithm using interval-valued fuzzy sets, Inf. Sci. 329 (2016) 144–163.
[55] G. David, A. Averbuch, Spectralcat: categorical spectral clustering of numerical and nominal data, Pattern Recognit. 45 (2012) 416–433.
[56] J. Wu, S. Pan, X. Zhu, Z. Cai, P. Zhang, C. Zhang, Self-adaptive attribute weighting for Naive Bayes classification, Expert Syst. Appl. 42 (2015) 1487–1502.
[57] L. Jiao, Q. Pan, T. Denœux, Y. Liang, X. Feng, Belief rule-based classification system: extension of FRBCS in belief functions framework, Inf. Sci. 309 (2015) 26–49.
[58] Y.-H. Shao, W.-J. Chen, Z. Wang, C.-N. Li, N.-Y. Deng, Weighted linear loss twin support vector machine for large-scale classification, Knowl.-Based Syst. 73 (2015) 276–288.
[59] S. Datta, S. Das, Near-bayesian support vector machines for imbalanced data classification with equal or unequal misclassification costs, Neural Netw. 70 (2015) 39–52.
[60] N. Landwehr, M. Hall, E. Frank, Logistic model trees, Mach. Learn. 59 (2005) 161–205.
[61] M. Fallahnezhad, M.H. Moradi, S. Zaferanlouei, A hybrid higher order neural classifier for handling classification problems, Expert Syst. Appl. 38 (2011) 386–393.