Self-Tuning Fuzzy Rule Bases with Belief Structure

Jun Liu[1], Da Ruan[2], Jian-Bo Yang[3], and Luis Martinez Lopez[4]

[1] School of Computing and Mathematics, University of Ulster at Jordanstown, Newtownabbey BT37 0QB, Northern Ireland, UK
j.liu@ulster.ac.uk
[2] Belgian Nuclear Research Centre (SCK•CEN), Boeretang 200, 2400 Mol, Belgium
[3] Manchester Business School (East), P.O. Box 88, The University of Manchester, Manchester M60 1QD, UK
[4] Dept. of Computer Sciences, University of Jaén, E-23071, Jaén, Spain

Abstract. A fuzzy rule-based evidential reasoning (FURBER) approach has been proposed recently, where a fuzzy rule-base designed on the basis of a belief structure (called *a belief rule base*) forms a basis in the inference mechanism of *FURBER*. This kind of rule-base with both subjective and analytical elements may be difficult to build in particular as the system increases in complexity. In this paper, a learning method for optimally training the elements of the belief rule base and other knowledge representation parameters in *FURBER* is proposed. This process is formulated as a nonlinear multi-objective function to minimize the differences between the output of a belief rule base and given data. The optimization problem is solved using the optimization tool provided in MATLAB. A numerical example is provided to demonstrate how the method can be implemented.

Key words: uncertainty, fuzzy logic, belief rule-base, evidential reasoning, optimization, MATLAB, safety estimate

1 Introduction

In real world applications, intrinsically vague information may coexist with conditions of "lack of specificity" originating from evidence not strong enough to completely support a hypothesis but only with degrees of belief or *credibility* [8]. One realistic way to deal with imprecision is to use linguistic assessments instead of numerical values. Fuzzy logic approaches [24] and [25] employing fuzzy IF-THEN rules can model the qualitative aspects of human knowledge and reasoning process without employing precise quantitative analysis. This actually provides a tool for working directly with the linguistic information, which are commonly used in engineering system analysis (e.g., representing

Jun Liu et al.: *Self-Tuning Fuzzy Rule Bases with Belief Structure*, Studies in Computational Intelligence (SCI) **5**, 419–437 (2005)
www.springerlink.com

risk factors and carrying out safety assessments) [1, 3, 7, 16]. Dempster-Shafer (D-S) theory of evidence [6, 13] based on the concept of *belief function* is well suited to modeling subjective credibility induced by partial evidence [15]. It also provides appropriate methods for computing belief functions for combination of evidence. Besides, the D-S theory also shows great potentials in multiple attribute decision analysis (MADA) under uncertainty, where an evidential reasoning (ER) approach for MADA under uncertainty has been developed, on the basis of a distributed assessment framework and the evidence combination rule of the D-S theory [18, 19, 20, 21, 22].

Accordingly, it seems reasonable to extend the fuzzy logic framework to cover credibility uncertainty as well. To combine fuzzy logic and D-S models to deal with fuzziness and incompleteness in safety analysis, a framework for modelling the safety of an engineering system using a FUzzy Rule-Based Evidential Reasoning (FURBER) approach has been recently proposed [9], which is based on a generic Rule-base Inference Methodology using the Evidential Reasoning approach (RIMER) proposed in [23]. Within this framework, a fuzzy rule-base designed on the basis of a belief structure, called *a belief rule base*, is used to capture uncertainty and non-linear relationships between the parameters, and the inference of the rule-based system is implemented using the evidential reasoning algorithm [21].

A belief rule base forms a basis in the inference mechanism of *FURBER*, which is a framework for representing expert knowledge but it is difficult to determine its elements entirely subjectively, in particular for a large scale rule base with hundreds of rules. Also, a change in a rule weight or an attribute weight may lead to significant changes in the performance of a belief rule base. Moreover, the form of fuzzy membership function in the antecedent of the rule still remains an important factor for the system performance.

As such, there is a need to develop a method that can generate an optimal belief rule base using expert judgments as well as statistical data. In this paper, a learning method for optimally training the elements of the belief rule base and other knowledge representation parameters in *FURBER* is proposed. This process is formulated as a *nonlinear objective function* to minimize the differences between the output of a belief rule base and given data and is solved using the optimization tool provided in MATLAB.

In addition, for some linguistic terms (such as *good*, *fair*, *intelligent*, *smart*, or *beautiful*, etc.), which do not have clearly defined bases, it is too subjective to clearly define them. If they appear in the consequent part of a rule, it is better to draw a conclusion which has the same linguistic values as the one in the consequent but with different degree of confidence. We do not try to change the linguistic values used in the consequent of the rule, but a degree of confidence of this conclusion is added. The optimization approach proposed in the present paper provides a more flexibility for optimally training, i.e., it is not required that the training sample should be numerical data pairs, which can be subjective judgment using linguistic values with belief. Consequently, the reasoning process can be approximately imitated by the optimized *FURBER* (e.g., assigning weights to antecedent attributes and/or by adjusting belief degrees in the consequents of rules in a systematic manner). This is one of the prominent features of the belief rule-base.

The rest of this paper is organized as follows. *FURBER* is briefly reviewed in Sect. 2. The optimization method for constructing belief rule-base in *FURBER* is proposed in Sect. 3. A numerical example is illustrated in Sect. 4. Conclusions are drawn in Sect. 5.

2 FURBER

This section reviews the *FURBER* framework [9].

2.1 Fuzzy Rule-Base with the Belief Structure

Fuzzy logic systems are knowledge-based or rule-based ones constructed from human knowledge in the form of fuzzy *IF-THEN* rules. For example, the following is a fuzzy *IF-THEN* rule for safety analysis [9]:

IF Failure Rate of a hazard is *frequent AND* Consequent Severity is *catastrophic AND* Failure Consequent Probability is *likely THEN safety estimate* is *Poor*.

To take into account the belief degrees of a rule, attribute weights and rule weights, fuzzy rules can be extended in the following way. In general, assume that the T antecedent parameters, $U_1, \ldots, U_T$ can be described by J_i linguistic terms $\{A_{ij}; j = 1, \ldots, J_i\}, i = 1, \ldots, T$, respectively. One consequent variable can be described by N linguistic terms, i.e., $D_1, D_2, \ldots, D_N$. Suppose that the rule-base is given by $R = \{R_1, R_2, \ldots, R_L\}$, the kth rule can be represented as follows:

$$R_k : \textit{IF}\, U \,\text{is}\, A^k\ \textit{THEN}\, D \,\text{with belief degree}\ \beta^k, \text{with a rule weight}\, \theta_k \,\text{and}$$
$$\text{attribute weights}\, \delta_1, \ldots, \delta_T \tag{1}$$

where U represents the antecedent attribute vector $(U_1, \ldots, U_T), A^k$ the packet antecedents $\{A_1^k, \ldots, A_T^k\}$, where $A_i^k (\in \{A_{ij}; j = 1, \ldots, J_i\})$ is a linguistic term corresponding to the ith attribute in the kth rule with $i = 1, \ldots, T. D$ is the consequent vector $(D_1, \ldots, D_N)$, and β^k the vector of the belief degrees $(\beta_{1k}, \ldots, \beta_{Nk})$ for $k \in \{1, \ldots, L\}$ with $\sum_{i=1}^{N} \beta_{ik} \leq 1$. This is the vector form of *a belief rule*, β_{ik} measures the degree to which D_i is the consequent if the input activates the antecedent A^k in the kth rule for $i = 1, \ldots, N$; $k = 1, \ldots, L$. L is the number of rules in the rule-base. If $\sum_{i=1}^{N} \beta_{ik} = 1$, the output assessment or the kth rule is said to be complete; otherwise, it is incomplete. The rule base in (1) is referred to as *a belief rule base*.

A belief rule base given in (1) represents functional mappings between antecedents and consequents with uncertainty. It provides a more informative

Table 1. A belief rule expression matrix

	Input					
Belief Output	A^1 (w_1)	A^2 (w_2)	$\ldots$	A^k (w_k)	$\ldots$	A^L (w_L)
D_1	β_{11}	β_{12}	$\ldots$	β_{1k}	$\ldots$	β_{1L}
$\vdots$	$\vdots$	$\vdots$	$\ldots$	$\vdots$	$\ldots$	$\vdots$
D_i	β_{i1}	β_{i2}	$\ldots$	β_{ik}	$\ldots$	β_{iL}
$\vdots$	$\vdots$	$\vdots$	$\ldots$	$\vdots$	$\ldots$	$\vdots$
D_N	β_{N1}	β_{N2}	$\ldots$	β_{Nk}	$\ldots$	β_{NL}

and realistic scheme for uncertain knowledge representations. Note that the degrees of belief $\beta_{ik}(i = 1, \ldots, N; k = 1, \ldots, L)$, and the weights could be assigned directly by experts or more generally they may be trained and updated using dedicated learning algorithms if a priori or *up-to-date* information regarding the input and output of a rule-based system is available. Once such a belief rule-base is established, the knowledge contained in the belief rule base can be used to perform inference for given input. The rule base can be summarized using a belief rule expression matrix shown in Table 1.

2.2 Fuzzy Rule-Base Inference Mechanism Based on the Evidential Reasoning Approach

In the matrix, w_k is the activation weight of A^k, which measures the degree to which the kth rule is weighted and activated. The degree of activation of the kth rule w_k is calculated as:

$$\omega_k = \left(\theta_k * \prod_{i=1}^{T_k} \left(\alpha_i^k\right)^{\bar{\delta}_i}\right) \Bigg/ \left(\sum_{i=1}^{L} \left[\theta_i * \prod_{i=1}^{T_k} \left(\alpha_l^i\right)^{\bar{\delta}_l}\right]\right) \tag{2}$$

where $\bar{\delta}_i = \delta_i / (\max_{i=1,2,3} \{\delta_i\})$. α_i^k $(i = 1, \ldots, T_k)$ is the individual matching degree to which the input for U_i belongs to A_i^k of the ith individual antecedent in the kth rule, T_k is the number of antecedents involved in the kth rule.

For a given real input vector $\boldsymbol{a} = (a_1, \ldots, a_{T_k}) = \mu_{A_i^k}(a_i)$, here is the fuzzy membership function of the linguistic term A_i^k. Fuzzy membership functions can be applied in different forms depending on the system. The straight-line membership functions can be used due to its advantage of simplicity, such as the triangular membership function and trapezoidal membership function. Continuous and differentiable Gaussian function is used in this paper, i.e.,

$$\mu_{A_i^k}(a_i) = \exp\left(-\frac{1}{2}\left(\frac{a_i - c_i^k}{\sigma_i^k}\right)^2\right) , \tag{3}$$

where c_i^k is the central value of fuzzy membership function and σ_i^k is the variance at the central value.

Having represented each rule as (1), the ER approach can be directly applied to combine rules and generate final conclusions as follows. First, transform the degrees of belief β_{jk} for all $j = 1, \ldots, N, k = 1, \ldots, L$ into basic probability masses and then aggregate all the packet antecedents of the L rules to generate the combined degree of belief in each possible consequent D_j in D using the evidential reasoning (ER) algorithm [20, 21]. The final conclusion generated by aggregating the L rules, which are activated by the actual input vector $\boldsymbol{a} = \{a_t;\, t = 1, \ldots, T\}$ can be represented as follows

$$\{(D_j, \beta_j), j = 1, \ldots, N\} \tag{4}$$

The ER Recursive Algorithm used in [21, 22] has been equivalently transformed into the ER overall analytical algorithm [17]. Using this overall algorithm, the overall combined degree of belief β_j in D_j is generated as follows:

$$\beta_j = \frac{\mu * \left[\prod_{k=1}^{L}\left(\omega_k \beta_{j,k} + 1 - \omega_k \sum_{j=1}^{N} \beta_{j,k}\right) - \prod_{k=1}^{L}\left(1 - \omega_k \sum_{j=1}^{N} \beta_{j,k}\right)\right]}{1 - \mu * \left[\prod_{k=1}^{L}(1 - \omega_k)\right]} \tag{5}$$

where $j = 1, \ldots, N$, and

$$\mu = \left[\sum_{j=1}^{N}\prod_{k=1}^{L}\left(\omega_k \beta_{j,k} + 1 - \omega_k \sum_{j=1}^{N} \beta_{j,k}\right) - (N-1)\prod_{k=1}^{L}\left(1 - \omega_k \sum_{j=1}^{N} \beta_{j,k}\right)\right]^{-1}$$

Notice that it is the beliefs used in the belief structure and the activation weights that determine the actual performance of inference. The degree to which the final output can be affected is determined by the magnitude of the activation weight and the belief degrees in each rule. Therefore, the performance of inference can be improved if the following parameters in (5) are adjusted by autonomous learning.

(1) Rule weight $\theta_k (k = 1, \ldots, L)$ and attribute weights $\delta_1, \delta_2, \delta_3$;
(2) The degrees of belief $\beta_{ik} (i = 1, \ldots, N; k = 1, \ldots, L)$;
(3) The central value of fuzzy membership function c_i^k and the variance σ_i^k at the central value.

Notice that there are some constraint conditions on each parameter in the above formulation, which are described in the optimization formulation in the following section.

3 Optimal Learning Method for Belief Rule Bases in FURBER

3.1 Multiple Variable Constrained Nonlinear Optimization Problem

In this section, the learning algorithm is to be incorporated in the context *FURBER* model whose function is to search for optimal belief rule matrix and other knowledge representation parameters simultaneously.

Based on the formulation of the system output, the learning method includes a *constrained nonlinear optimization* problem. This objective can be formulated as the minimization of a nonlinear programming problem expressed as follows:

$$\begin{aligned} &\min . f(\boldsymbol{P}) \\ \text{s.t. } &A(\boldsymbol{P}) = 0 \\ &B(\boldsymbol{P}) \geq 0 \end{aligned} \tag{6}$$

where $f(\boldsymbol{P})$ is the objective function, $\boldsymbol{P}$ is the parameter vector or matrix of the system, $A(\boldsymbol{P})$ are the equality functions and $B(\boldsymbol{P})$ are the inequality functions respectively.

The optimization starts with the pre-selected initial values of parameters. Then the nonlinear algorithm is used to iteratively adjust the parameters, until the objective function (6) is minimized. In the proposed method, a *nonlinear objective function* is formulated to minimize the differences between the output of a belief rule base and given data. Parameter specific limits and partial expert judgments are formulated as constraints. These so determined parameters are the optimal settings of the *FURBER*.

In the learning process we must use observations on the input and output to determine the parameters. In the following, we shall assume that we have available a collection of observation pairs $(\boldsymbol{x}, y)$, where $\boldsymbol{x}$ is an input vector, and y is the corresponding output. Besides the ability to handle various types of input information, as one may see from the output of the *FURBER*, it also provides the flexibility of the output status, i.e., the qualitative output forms (subjective judgment).

Notice that the output of *FURBER* is actually a distribution assessment instead of a single numerical score, which provides a panoramic view about the output status, from which one can see the variation between the original output and the revised output on each linguistic term. A distribution is easier to understand and flexible to represent output information than a single average value. Especially it is very useful in the case that the outputs are too subjective to quantify them. In fact, this subjective judgment with belief is quite popular in such area as diagnosis, classification, prediction etc. For instance, a physician wishes to predict the likelihood of a new patient's falling ill given his symptoms by analyzing a database of previous patient symptoms and outcomes. Teachers wish to determine which students need extra help given their aptitude test score by analyzing a database of past student test scores and performance. Manufactures wish to determine which parts will fail under stress by analyzing a database of manufacturing parameters and previous part failures and so on.

Hence, for example, for some linguistic terms (such as *good*, *fair*, *intelligent*, *smart*, or *beautiful*, etc.), which do not have clearly defined bases, it is too subjective to clearly define them. If they appear in the consequent part of a rule, it is better to draw a conclusion which has the same linguistic values as the one in the consequent but with a different degree of confidence, i.e., we do not try to change the linguistic values used in the consequent of the rule, but a degree of confidence of this conclusion is added. The optimization approach proposed in the present paper provides more flexibility for optimal training, i.e., it is not required that the training sample should be numerical data pairs, which can be subjective judgment using linguistic values with belief.

3.2 The Optimization Algorithm Based on the Output in the Form of the Subjective Judgment

In this case, a training set composed of M input-output pairs $(x_m, y_m)(m = 1, \ldots, M)$, where y_m can be a subjective judgment, i.e., a distributed assessment on the linguistic value with belief. Notice that the single judgment as one linguistic value can be regarded as a special case of the distribution assessment.

In this paper the Matlab Optimization Toolbox is applied [5]. Since the function to be minimized and the constraints are both continuous, for this case based on the output in the form of the subjective judgment is to solving problems with multi-objective functions using Fminimax in Matlab [5] while the output is given as the expert judgment using a belief distribution. Based on (4), i.e., the final conclusion generated by aggregating the L rules, which are activated by the actual input can be represented as follows

$$\{(D_j, \beta_j), j = 1, \ldots, N\} \tag{7}$$

So each $(D_j, \beta_j)(j \in \{1, \ldots, N\})$ can be regarded as one component of the multi-objective function vector. The function to be minimized is continuous. This multi-objective function is solved using Fminimax function in MATLAB referred to as the *minimax problem* defined as follows:

$$\min_{Q} \max_{\{\xi_j\}} \{\xi_j(\boldsymbol{Q}); j = 1, \ldots, N\} \tag{8}$$

where

$$\xi_j(\boldsymbol{Q}) = \frac{1}{M} \sum_{m=1}^{M} (\beta_j(m) - \hat{\beta}_j(m))^2, j = 1, \ldots, N \tag{9}$$

Here $\boldsymbol{Q}$ is the tuning parameter vector. $\beta_j(m)$ is given by (5) for the mth input in training set. M is the number of points in the training set, $\hat{\beta}_j(m)$ is the expected belief corresponding to the individual consequent D_j.$(\beta_j(m) - \hat{\beta}_j(m))$ is the residual at the mth point. The tuning parameters are beliefs, weights and parameters of fuzzy membership function, without utilities. Equation (8) is an N-objective and multi-variable nonlinear optimization problem. The constraint conditions are given as follows:

$$0 \leq \beta_{jk} \leq 1, j = 1, \ldots, N; k = 1, \ldots, L \tag{10a}$$

$$\textstyle\sum_{j=1}^{N} \beta_{jk} = 1 \tag{10b}$$

$$0 \leq \delta_i \leq 1, i = 1, \ldots, T \tag{10c}$$

$$0 \leq \theta_k \leq 1, k = 1, \ldots, L \tag{10d}$$

$$lb_{MF} \leq c_{ij} \leq ub_{MF}, i = 1, \ldots, 3; j = 1, \ldots, J_i \tag{10e}$$

$$c_{ij} \leq c_{ik} \ \text{ if } \ j \leq k, j, k = 1, \ldots, J_i \tag{10f}$$

$$0 \leq \sigma_{ij}, i = 1, \ldots, 3; j = 1, \ldots, J_i \tag{10g}$$

lb_{MF} and ub_{MF} are the bounds of the universal courses of fuzzy membership function. Here, $\sum_{j=1}^{N} \beta_{jk} = 1$, i.e., the optimized rule-base should be complete. Therefore, each generation of the optimization algorithm is used to get the minimal mean square error. There is no nonlinear constraint here, all the constraints are linear. Arrange the problem into the standard form of MATLAB, and use the FMINCON function to solve the problem.

Minimax method [5, 11, 12] is also called *ideal point method* that is to minimize a worst case objective function. In other words, the purpose of minimax formulation strategy is to minimize the maximum relative deviation of the objective function from its minimum objective function value. All the objective functions are evaluated qualitatively according to their functional importance and are assigned weights, $\boldsymbol{\omega} = (\omega_{1,\ldots,}\omega_N)$ indicating the designer's subjective preference. Usually $0 \leq \omega_j \leq 1$, $\sum_{j=1}^{N} w_j = 1$. The group of multiple objective function is separated and the optimum is $\boldsymbol{P}^*$ in feasible solution space corresponding to each objective function vector $\xi^*(\boldsymbol{Q}) = \{\xi_j^*(\boldsymbol{Q});$ $j = 1, \ldots, N\}$. Specify $\xi_j^*\ (\boldsymbol{Q}) = \xi_j^*$. So the objective function can be formulated as follows:

$$\min_{Q} \max_{j=1,\ldots,N} \{\varphi_j(\boldsymbol{Q})\} \tag{11}$$

with

$$\varphi_j(\boldsymbol{Q}) = \frac{\xi_j(\boldsymbol{Q}) - \xi_j^*}{\xi_j^*}, \xi_j^* > 0, j = 1, \ldots, N \tag{12}$$

The computational steps of the minimax method of a multi-objective optimization problem in (11) are summarized below:

Step 1. Based on the multi-objective function, solve the following single-objective optimization problem with the conventional methods individually (e.g., FMINCON).

$$\begin{aligned} &\min_{Q} \xi_j(\boldsymbol{Q}) \\ &\text{s.t. } 10(a) \sim 10(g) \end{aligned} \tag{13}$$

where $j = 1, \ldots, N$, $\xi_j\ (\boldsymbol{Q})$ is given by (9), and there are N single-objective optimization problem to solve. Suppose that the optimum is $\boldsymbol{Q}^*$ in feasible solution space and its corresponding function value $\boldsymbol{\xi}^* = \{\xi_j^*; j = 1, \ldots, N\}$.

Step 2. To gain the designer's preference of the objectives, a relative weight vector $\omega = (\omega_{1,\ldots,}\omega_N)$ is given. Here each component (D_j, β_j) of the multi-objective function is regarded as having the same weight, i.e., equally important.

Step 3. The multi-objective function can be formulated as follows:

$$\min_{\boldsymbol{Q}} \max_{j=1,\ldots,N} \left\{ \omega_j \cdot \frac{\xi_j(\boldsymbol{Q}) - \xi_j^*}{\xi_j^*} \right\}$$
$$\text{s.t.}, 10(a) \sim 10(g) \tag{14}$$

Step 4. Arrange the problem into the standard form of MATLAB, and use the MATLAB function, FMINIMAX, to calculate the result.

This optimization process proposed is a kind of iterative process, the iterative process continues until the mean square error becomes smaller than a specified tolerance which can be determined by the expert within the relevant application context.

4 A Numerical Example

The *FURBER* framework has been applied in [9] modeling system safety of an offshore and marine engineering system: floating production storage offloading (FPSO) system [4, 10], specially focus on collision risk between FPSO and a shuttle tanker due to technical failure during a tandem offloading operation.

4.1 Problem Description and the Optimization Algorithms

In this section, an example within the same application framework in [9] is used to demonstrate how the optimization method can be implemented in safety analysis. As an illustration, we only consider the safety assessment related to controllable pitch propeller (CPP) failure to demonstrate the procedure involved in the optimization of belief rule base and other knowledge representation parameters in *FURBER*.

The three fundamental parameters used to assess the safety level of an engineering system on a subjective basis are the *failure rate* (**FR**), *consequence severity* (**CS**) and *failure consequence probability* (**FCP**). Subjective assessments (using linguistic variables instead of ultimate numbers in probabilistic terms) are more appropriate for safety analysis as they are always associated with great uncertainty, especially for a novel system with high level of innovation. These linguistic assessments can become the criteria for measuring safety levels. The typical linguistic variables used to describe **FR, CS, FCP** of a particular element may be described as follows [8, 14]:

FR describes failure frequencies in a certain period, which directly represents the number of failures anticipated during the design life span of a particular system or an item. To estimate the **FR**, one may choose to use such linguistic terms as "*very low,*" "*low,*" "*reasonably low,*" "*average,*" "*reasonably frequent,*" "*frequent,*" and "*highly frequent.*"

CS describes the magnitude of possible consequences, which is ranked according to the severity of failure effects. One may choose to use such linguistic terms as "*negligible,*" "*marginal,*" "*moderate,*" "*critical,*" and "*catastrophic.*"

FCP defines the probability that consequences happened gives the occurrence of the event. One may choose to use such linguistic terms as "*highly unlikely,*" "*unlikely,*" "*reasonably unlikely,*" "*likely,*" "*reasonably likely,*" and "*Definite.*"

For the detailed definitions of these parameters we refer to [8, 14].

Safety estimate is the only output fuzzy variable used in this study to produce safety evaluation for a particular cause to technical failure. This variable is also described linguistically, which is described and determined by the above parameters. In safety assessment, it is common to express a safety level by degrees to which it belongs to such linguistic variables as "*poor,*" "*fair,*" "*average,*" and "*good*" that are referred to as safety expressions. These linguistic terms do not have clearly defined bases and it is difficult to clearly quantitatively define them. They appear in the consequent part of a rule, it is better to draw a conclusion which has the same linguistic values as the one in the consequent but with different degree of confidence, i.e., a degree of confidence of this conclusion is added. This subjective judgment with belief is popular and useful in such area as safety/risk classification.

Twenty-seven fuzzy rules from a total of 245 rules [9, 14] are extracted and used in our example, which are described in Appendix. They are used as the initial belief rule base in the learning process. Here the linguistic terms for describing **FR** are supposed to be (*average*, *frequent*, *highly frequent*), for **CS** (*moderate*, *critical*, *catastrophic*), and for **FCP** (*likely*, *highly likely*, *definite*) respectively. The definitions of their linguistic terms refer to [9, 14], and the corresponding initial membership function are given by Gaussian function as shown in Fig. 1 in Sect. 4.2.

Here it is assumed that each input parameter may be fed to the proposed safety model in a single deterministic value although there are other input forms possible to address the inherent uncertainty associated with the data as discussed in [9, 14].

Notice that here the training data are given in subjective output forms. So the corresponding optimization formulation is given in (8) for subjective judgment form, where $L = 27, T = 3, N = 4$. The computation steps are given in Sect. 3. The FMINIMAX function is used to solve the problem.

4.2 Experiment Results Base on Distributed Assessment with Belief

For the learning purpose, a set of 14 data is used. What we would like to retrieve from the learning are the tuned and refined beliefs, weights, and parameters of fuzzy membership functions. Here we partition the available data into a training set and a test set commonly tests generalization. The training set is used for parameter training purposes; and, once a goal performance measure value is achieved, the corresponding approximation error on the test data is measured. In the example, the output is given by the subjective judgment, i.e., a distribution assessment with belief.

We use this example as an illustration, i.e., 7 training data for parameter estimation (Table 2). 7 remaining data for test purpose (Table 4). The initial belief matrix and the rule weights are supposed to be given by Expert as shown in [9]. The initial attribute weights are all equal to 1.

Table 3 shows the comparison between the expected distributed assessment and the learning results based on distributed assessment with belief from expert. In this example, the error tolerance for each sub-objective minimization is set to 0.00001 and the maximum iteration is set to 100, and the error tolerance for multi-objective minimization is set to 0.0001 and the maximum iteration is set to 100 to avoid an endless loop in the learning process.

Table 2. 7 training input vaues of three safety related antecedent attributes

	Antecedents		
Number	FR	CS	FCP
1	7.75	8.25	7.6
2	7	8	7.25
3	8	8.5	7
4	7	7	5.5
5	6.5	8	7.5
6	7.15	7.95	7.25
7	7	8.5	7

In Table 3 (in Table 5 as well), G represents "*Good;*" A represents "*Average;*" F represents "*Fair;*" P represents "*Poor,*" respectively.

The initial belief structure of the rule base is given by experts shown in Appendix. The initial fuzzy membership function and the optimized fuzzy membership function are shown in Fig. 1 and Fig. 2 respectively.

After the tuning, a testing data consisting of 7 records (Table 4) is used. The expected and actual values of the output are listed in Table 5.

As one may see from Table 3, there is a big difference between the initial and expected outputs. The big difference is due to the fact that the initial outputs are obtained when the belief rule matrix and the weights have not been tuned. Although the difference is not great, that may be due to the small size of the rule base, however, it would be difficult to obtain such relatively accurate belief rule matrix while the system increases in complexity.

Table 3. Training results comparison based on distributed assessment with belief

	Safety Estimate (Distribution Assessment with Belief)			
	Expected Values of Output			
Number	G	A	F	P
1	0	0.0123	0.3641	0.6236
2	0	0.0033	0.3090	0.6876
3	0	0.0057	0.3735	0.6208
4	0	0.0373	0.7802	0.1825
5	0	0.0640	0.4165	0.5195
6	0	0.0013	0.4179	0.5808
7	0	0.0047	0.6151	0.3802
	Trained Output Values			
Number	G	A	F	P
1	0.0098	0.3462	0.6342	0.0098
2	0.0419	0.4510	0.4652	0.0419
3	0.0060	0.3374	0.6506	0.0060
4	0.0016	0.7677	0.2292	0.0016
5	0.0502	0.4689	0.4307	0.0502
6	0.0359	0.4364	0.4918	0.0359
7	0.0395	0.3503	0.5707	0.0395
	Initial System Output Values Before Training			
Number	G	A	F	P
1	0	0	0.8056	0.1944
2	0	0	0.8310	0.1690
3	0	0	0.7982	0.2018
4	0	0	0.9768	0.0232
5	0	0	0.7808	0.2192
6	0	0	0.8473	0.1527
7	0	0	0.7218	0.2782

Especially, as shown in Table 3 and Table 5, after the parameter optimization, the performance of the system is perfectly achieved. One may notice that the testing results in the above Table indicate that 90% of them are already correct within the specified tolerance 0.01 (except test 6). The error including test 6 would be decreasing while the number of training data is increasing.

As a result, a belief-rule-base matrix, and other knowledge representation parameters can be built from partial knowledge about the output and can be then refined by learning, or it can begin with an empty rule base (randomly generated) that is filled by creating rule-bases from the training data. The optimized belief rule base is given in Table A1 of Appendix.

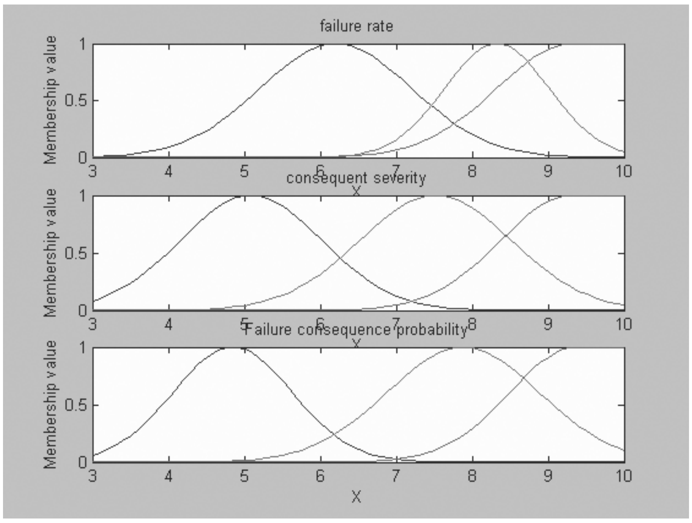

Fig. 1. The initial fuzzy membership function

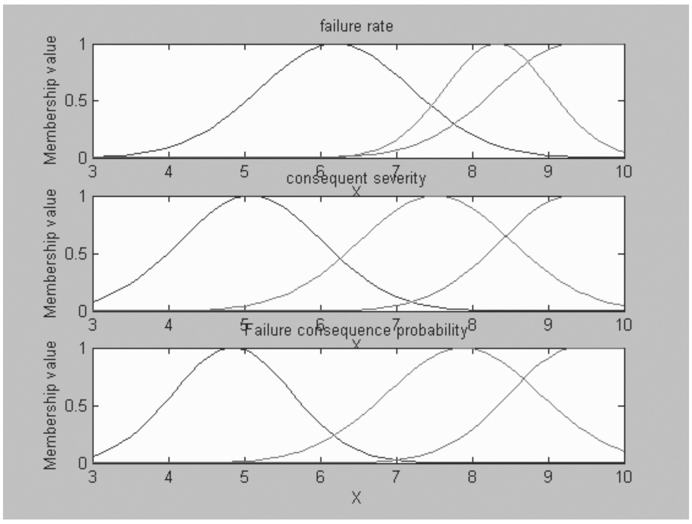

Fig. 2. The optimized fuzzy membership function for the case based on the subjective output

Table 4. 7 test input vaues of three safety related antecedent attributes

	Antecedents		
Number	FR	CS	FCP
1	7.5	8	6
2	7.5	7.2	7.1
3	7.5	8.5	7
4	7	7	6
5	8	8.5	7.5
6	7.25	6.75	7
7	7.95	8.25	7.9

Table 5. Test results comparison based on distributed assessment with belief

	Safety Estimate (Distribution Assessment with Belief)			
	Expected Values of Output			
Number	G	A	F	P
1	0	0.0041	0.6142	0.3817
2	0	0.0080	0.3694	0.6226
3	0	0.0102	0.3595	0.6303
4	0	0.0097	0.6926	0.2977
5	0	0.0097	0.3930	0.5973
6	0	0.0200	0.5733	0.4067
7	0	0.0256	0.2688	0.7056
	Trained Output Values			
Number	G	A	F	P
1	0.0062	0.6956	0.2920	0.0062
2	0.0182	0.3818	0.5818	0.0182
3	0.0182	0.3561	0.6075	0.0182
4	0.0102	0.7069	0.2726	0.0102
5	0.0059	0.3386	0.6497	0.0059
6	0.0278	0.4116	0.5328	0.0278
7	0.0060	0.3498	0.6383	0.0060
	Initial System Output Values before Training			
Number	G	A	F	P
1	0	0	0.9602	0.0398
2	0	0	0.8684	0.1316
3	0	0	0.8084	0.1916
4	0	0	0.9632	0.0368
5	0	0	0.7756	0.2244
6	0	0	0.8050	0.1950
7	0	0	0.7799	0.2201

5 Conclusion

A learning method for optimally training a fuzzy rule base with the belief structure in FURBER for engineering system safety analysis is proposed. This learning method provides practical support to construct flexible and reliable belief rule bases, which can optimally imitate complex reasoning processes and represent nonlinear or nonsmooth relationships using both human knowledge and numerical data. The unique feature of the new method is that the output data can be judgmental, which makes the learning process flexible and practical in decision making.

Acknowledgements

This work forms part of the projects supported by the UK Engineering and Physical Sciences Research Council (EPSRC) under Grant No: GR/S85498/01 and Grant No: GR/S85504/01.

Appendix

Table A1. The optimized belief rule base

	Trained Belief Structure of the Rule			
Rule Number	Good	Average	Fair	Poor
1	0.0020	0.0001	0.9434	0.0545
2	0.0155	0.0250	0.5058	0.4537
3	0.0013	0.0003	0.8364	0.1620
4	0.0108	0.0038	0.9852	0.0001
5	0.2144	0.1123	0.0230	0.6503
6	0.0004	0.0000	0.0000	0.9996
7	0.0604	0.0073	0.9323	0.0000
8	0.2699	0.1107	0.0256	0.5939
9	0.0005	0.0000	0.0000	0.9995
10	0.0000	0.0000	0.9996	0.0004
11	0.0002	0.0002	0.9820	0.0177
12	0.0002	0.0015	0.0002	0.9981
13	0.0231	0.0010	0.9717	0.0042
14	0.0323	0.0135	0.0043	0.9499
15	0.0635	0.0215	0.0163	0.8986
16	0.0537	0.0186	0.0002	0.9274
17	0.2584	0.1029	0.0013	0.6374
18	0.0916	0.0391	0.0033	0.8660
19	0.0000	0.0000	1.0000	0.0000
20	0.0000	0.0001	0.9977	0.0022
21	0.0000	0.0000	0.0000	1.0000
22	0.0030	0.0001	0.9959	0.0010
23	0.0450	0.0159	0.0010	0.9381
24	0.0796	0.0312	0.0011	0.8882
25	0.0066	0.0025	0.0000	0.9908
26	0.0780	0.0271	0.0009	0.8941
27	0.0865	0.0287	0.0003	0.8845

For example, the first rule (the third row) in this table represents the following rule:

Rule # 137: *IF the* ***failure rate*** *is average AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is* {(*good*, 0.0020), (*average*, 0.001), (*fair*, 0.9434), (*poor*, 0.0545)}

RULE BASE using in the case study which is extracted from the rule-base established in [14]

1. Rule # 137: *IF the* ***failure rate*** *is average AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
2. Rule # 139: *IF the* ***failure rate*** *is average AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is poor*
3. Rule # 140: *IF the* ***failure rate*** *is average AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is poor*
4. Rule # 193: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is moderate AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
5. Rule # 195: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is moderate AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is fair*
6. Rule # 196: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is moderate AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is poor*
7. Rule # 200: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is critical AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
8. Rule # 202: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is critical AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is poor*
9. Rule # 203: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is critical AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is poor*
10. Rule # 207: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is poor*
11. Rule # 209: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is poor*

12. Rule # 210: *IF the* ***failure rate*** *is frequent AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is poor*
13. Rule # 214: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is negligible AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
14. Rule # 216: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is negligible AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is fair*
15. Rule # 217: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is negligible AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is fair*
16. Rule # 221: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is marginal AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
17. Rule # 223: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is marginal AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is fair*
18. Rule # 224: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is marginal AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is fair*
19. Rule # 228: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is moderate AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
20. Rule # 230: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is moderate AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is fair*
21. Rule # 231: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is moderate AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is poor*
22. Rule # 235: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is critical AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is fair*
23. Rule # 237: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is critical AND the* ***failure consequence probability*** *is highly likely THEN the* ***safety estimate*** *is poor*
24. Rule # 238: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is critical AND the* ***failure consequence probability*** *is definite THEN the* ***safety estimate*** *is poor*
25. Rule # 242: *IF the* ***failure rate*** *is highly frequent AND the* ***consequence severity*** *is catastrophic AND the* ***failure consequence probability*** *is likely THEN the* ***safety estimate*** *is poor*
26. Rule # 244: IF the **failure rate** is *highly frequent* AND the **consequence severity** is *catastrophic* AND the **failure consequence probability** is *highly likely* THEN the **safety estimate** is *poor*

27. Rule # 245: IF the **failure rate** is *highly frequent* AND the **consequence severity** is *catastrophic* AND the **failure consequence probability** is *definite* THEN the **safety estimate** is poor

References

1. Bell P.M. and Badiru A.B. (1996), Fuzzy modelling and analytic hierarchy processing to quantify risk levels associated with occupational injuries – Part I: The development of fuzzy-linguistic risk levels, *IEEE Transactions on Fuzzy Systems*, Vol. 4, No. 2, pp. 124–131. 420
2. Binaghi E. and Madella P. (1999), Fuzzy Dempster–Shafer reasoning for rule-based classifiers, *International Journal of Intelligent Systems*, Vol. 14, pp. 559–583.
3. Bowles J.B. and Pelaez C.E. (1995), Fuzzy logic prioritisation of failures in a system failure mode, effects and criticality analysis, *Reliability Engineering and System Safety*, Vol. 50, pp. 203–213. 420
4. Chen H. and Moan T. (2002), Collision risk analysis of FPSO-tanker offloading operation, *21st International Conference on Offshore Mechanics & Arctic Engineering*, 23–28 June 2002, Oslo, Norway. 427
5. Coleman T., Branch M.A, and Grace A. (1999), *Optimization Toolbox – for Use with Matlab*, The Mathworks Inc. 425, 426
6. Dempster A.P. (1968), A generalization of Bayesian inference, *Journal of the Royal Statistical Society, Series B*, Vol. 30, pp. 205–247. 420
7. Duckstein L. (1994), Elements of fuzzy set analysis and fuzzy risk, In *Decision Support Systems in Water Resources Management*, H.P. Nachtnebel (Ed.), pp. 410–430. Paris: UNESCO Press. 420
8. Liu J., Yang J.B., Wang J. and Sii H.S. (2003), Review of uncertainty reasoning approaches as guidance for maritime and offshore safety-based assessment, *J. of UK Safety and Reliability Society*, Vol. 23, No. 1, pp. 63–80. 419, 427, 428
9. Liu J., Yang J.B., Wang J., Sii H.S., Wang Y.M. (2004), Fuzzy rule-based evidential reasoning approach for safety analysis, *International Journal of General Systems*, **33** (2–3), 183–204. 420, 421, 427, 428, 429
10. McCaul J.R. (2001), Special Report – Floating production systems, *Oil & Gas Journal*, June, 11. 427
11. Marler R.T. and Arora J.S. (2004), Survey of multi-objective optimization methods for engineering, *Struct Multidisc Optim.*, Vol. 26, pp. 369–395. 426
12. Miettinen K. (1999), *Nonlinear Multiobjective Optimization*. Boston: Kluwer Academic Publishers 426
13. Shafer G. (1976), *A Mathematical Theory of Evidence*. Princeton, N.J.: Princeton University Press. 420
14. Sii H.S. and Wang J. (2002), *Safety assessment of FPSOs – The process of modelling system safety and case studies*, Report of the project – "*The Application of Approximate Reasoning Methodologies to Offshore Engineering Design*" (EPSRC GR/R30624 and GR/R32413), Liverpool John Moores University, UK. 427, 428, 434
15. Smets P. (1988), Belief function, in *Non-Standard Logics for Automated Reasoning* (Smet P., Mamdani E.H., Dubois D., and Prade H. eds.), London, Academic Press, pp. 253–277. 420

16. Wang J., Yang J.B. and Sen P. (1995), Safety analysis and synthesis using fuzzy set modelling and evidential reasoning, *Reliability Engineering and System Safety*, Vol. 47, No. 3, pp. 103–118. 420
17. Wang Y.M., Yang J.B. and Xu D.L. (2003), Environmental impact assessment using the evidential reasoning approach, accepted by *European Journal of Operational Research*. 423
18. Yang J.B. and Singh M.G. (1994), An evidential reasoning approach for multiple attribute decision making with uncertainty, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 24, No. 1, pp. 1–18. 420
19. Yang J.B. and Sen P. (1994), A general multi-level evaluation process for hybrid MADM with uncertainty, *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 24, No. 10, pp. 1458–1473. 420
20. Yang J.B. (2001), Rule and utility based evidential reasoning approach for multi-attribute decision analysis under uncertainties. *European Journal of Operational Research,* Vol. 131, pp. 31–61. 420, 423
21. Yang J.B. and Xu D.L. (2002a), On the evidential reasoning algorithm for multiple attribute decision analysis under uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics-Part A*, Vol. 32, No. 3, 289–304. 420, 423
22. Yang J.B. and Xu D.L. (2002b), Nonlinear information aggregation via evidential reasoning in multi-attribute decision analysis under uncertainty. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, Vol. 32, No. 3, pp. 376–393. 420, 423
23. Yang J.B., Liu J., Wang J., and Sii H. S. (2003), A generic rule-base inference methodology using the evidential reasoning approach – RIMER, *IEEE Transactions on Systems, Man, and Cybernetics*, accepted and in press. 420
24. Zadeh, L. A. (1965), Fuzzy sets, *Information and Control*, Vol. 8, pp. 338–353. 419
25. Zadeh L.A. (1975), The concept of a linguistic variable and its applications to approximate reasoning. Part I, *Information Sciences*, Vol. 8, pp. 199–249; Part II, *Information Sciences*, Vol. 8, pp. 301–357; Part III, *Information Sciences*, Vol. 9, pp. 43–80. 419