A Learning Mechanism for BRBES Using Enhanced Belief Rule-Based Adaptive Differential Evolution

Raihan Ul Islam
Pervasive and Mobile Computing Laboratory
Luleå University of Technology, Sweden
Email: raihan.ul.islam@ltu.se

Mohammad Shahadat Hossain
Department of Computer Science and Engineering
University of Chittagong, Bangladesh
Email: hossain_ms@cu.ac.bd

Karl Andersson
Pervasive and Mobile Computing Laboratory
Luleå University of Technology, Sweden
Email: karl.andersson@ltu.se

***Abstract*—Nowadays, belief rule-based expert systems (BRBESs) are widely used in various domains which provides a framework to handle qualitative and quantitative data by addressing several kinds of uncertainty. Learning plays an important role in BRBES to upgrade its knowledge base and parameters values, necessary for the improvement of the prediction accuracy. Different optimal training procedures such as Particle Swarm Optimisation (PSO), Differential Evolution (DE), and Genetic Algorithm (GA) have been used as learning mechanisms. Among these procedures, DE performs comparatively better than others. However, DE's performance depends significantly in assigning near optimal values to its control parameters including cross over and mutation factors. Therefore, the objective of this article is to present a novel optimal training procedure by integrating DE with BRBES. This is named as enhanced belief rule-based adaptive differential evolution (eBRBaDE) algorithm because it has the ability to determine the near-optimal values of both the control parameters while ensuring the balanced exploitation and exploration in the search space. In addition, a new joint optimization learning mechanism by using eBRBaDE is presented where both parameter and structure of BRBES are considered. The reliability of the eBRBaDE has been compared with evolutionary optimization algorithms such as GA, PSO, BAT, DE and L-SHADE. This comparison has been carried out by taking account of both conjunctive and disjunctive BRBESs while predicting the Power Usage Effectiveness (PUE) of a datacentre. The comparison demonstrates that the eBRBaDE provides higher prediction accuracy of PUE than from other evolutionary optimization algorithms.**

***Contribution*—An enhanced differential evolution algorithm has been proposed in this paper, which is later used as a novel optimal training procedure for BRBES.**

***Index Terms*—Optimization, Learning, Evolutionary Algorithm, Differential evolution, Belief Rule-based Expert Systems.**

I. Introduction

Belief rule-based expert system (BRBES) facilitates the representation of uncertain knowledge by incorporating belief structure. Usually, an expert system comprises two main components one is the knowledge base, and the other is the inference engine. Human knowledge is usually represented using IF-THEN rules, e.g. "IF the presence of creatinine THEN renal failure is definite". This rule implies with 100% certainty that if there is a presence of creatinine, there will be renal failure. However, in the real-world scenario, this cannot be concluded with 100% certainty. Hence IF-THEN rule is limited in representing this kind of uncertain knowledge; rather, it represents assertive knowledge. Yang et al. [1] proposed a new knowledge representation schema by incorporating a belief structure in the consequent part of an IF-THEN rule. For example, "IF the amount of rainfall is Medium and duration of rain is High THEN chance of flooding is (High 60%, Medium 30%, Low 10%)". From this example, the belief structure incorporated with the consequent part of the rule can be understood. This belief structure is used to represent the uncertainty as will be elaborated in Subsection III-B. Evidential reasoning is used as an inference engine in the BRBESs, which has the capability of handling uncertainty in the reasoning process [1]. Because of these features the BRBESs have been used in different domains such as natural disaster prediction [2], health informatics [3], [4], and e-governance [5] where the issue of uncertainty is dominant in making the decision. The Belief Rule Base (BRB), consisting of belief rules, is called the knowledge base of BRBES. It is of two types, one is called conjunctive and the other is called disjunctive. In the conjunctive BRB the elements of the antecedent of a rule are connected with conjunctive operator. However, in the disjunctive BRB the elements of the antecedent part of the rule are connected with disjunct operator. However, conjunctive BRB suffers from the combinatorial explosion problem due to the conjunctive nature of the rule. Disjunctive BRB does not suffer from the combinatorial problem; hence it requires less computational time [6].

The performance of BRBES can be improved by obtaining optimal values of its various learning parameters including rule weight, attribute weights, and belief degrees of the belief structure by using various training procedures. This training procedure can be considered as a learning mechanism of BRBES. Usually, these learning parameters are set by the experts, which may not be accurate for large datasets, pro-

978-1-7281-9331-1/20/$31.00

ducing imprecise results. Therefore, to improve the accuracy of the results of BRBESs, different optimal training procedures have been used to support the learning mechanisms [7], [8], [9]. Different optimal training procedures (such as Sequential Quadratic Programming, Particle Swarm optimisation (PSO), and Differential Evolution) have been used as learning mechanisms in BRBES. Furthermore, the learning mechanisms of BRBES can be divided into two types. One is structure optimisation, referring to the identification of the optimal number of referential values of the rules of a BRB. The other is parameter optimisation, which refers to finding the optimal values of the learning parameters. However, the integration of both structure and parameter optimisations, known as joint optimisation, produces better results [10].

Sequential quadratic based optimal training procedures are usually used for parameter optimisation, which tends to get stuck in local optima and hence, unable to find the global optima in the search space. However, this limitation has been overcome by the evolutionary algorithms such as Particle Swarm Optimization (PSO) and Differential Evolution (DE), since they allow searching randomly from any point in the search space.

Among various evolutionary algorithms, DE performs better as an optimal training procedure for BRBES due to its optimisation strategy [10]. Exploitation and exploration are the two main components of the DE. Exploitation is the process of generating a new solution using the information from the current focus area of the search space [11]. Exploration helps to explore the search space more, which helps in avoiding getting stuck in local optima. However, the control parameters (such as mutation (F) and crossover (CR) factors) play a significant role in finding the global optima of a solution [12]. Determining the near right values of F and CR are very difficult and challenging [13]. Usually trial-and-error based procedure is used to find the optimal values of the control parameters, which is time-consuming. However, there has been growing interest in developing algorithms to predict the near right values of control parameters of DE [13].

Therefore, this research presents a parameter optimisation algorithm by integrating DE with BRBES, named enhanced BRB adaptive DE (eBRBaDE) to find the optimal values of the control parameters. Additionally, a learning mechanism consists of eBRBaDE for BRBES is presented. To evaluate, the effectiveness of the proposed learning mechanism, eBRBaDE has been used for predicting power usage effectiveness (PUE) for data centres, where the optimised BRBES shows better performance than other learning mechanisms.

The rest of the paper is organized as follows. Section 2 discusses the related works on different learning mechanisms of BRBES and DE algorithm. Section 3 describes the methodology of the new learning approach for BRBES based on joint optimisation using optimal training procedure named, eBRBaDE. Section 4 present the results, while section 5 concludes the paper and indicates future work.

II. Related Work

In this section, the scientific works related to learning in BRBES and different variants of DE are discussed.

Yang et al. [7] first proposed an optimisation model for BRBES. The authors suggested optimisation of the parameters using the non-linear constrained solver, named *fmincon* of the MATLAB optimisation toolbox. Sequential quadratic programming algorithm is used in *fmincon* solver, which is a deterministic algorithm. The proposed method is prone to get stuck in local optima due to its gradient-based mechanism. Additionally, for large numbers of variable the convergence rate is slow, which requires more time to find optimal solution. Zhou et al. [14] discussed the limitation of the *fmincon* based optimisation method and mentioned that this method is influenced by the initialization of the variables. Furthermore, Zhou et al. [14] suggested of using evolutionary algorithms due to their success in achieving the optimal or near-optimal solution for problems with nonlinear and continuous search space. In the aforementioned article, clonal selection algorithm (CSA) was used for optimising BRBES for diagnostic of lymph node metastasis in gastric cancer. In addition, other evolutionary algorithms, like PSO [15], DE [8], [9] were also used for optimisation of BRBES, where DE performed better than the others.

Yang et al. [9] also proposed a joint optimisation model for BRBES. A generalization error based on Hoeffding inequality theorem was proposed instead of root mean square error. A heuristic algorithm was used for structure optimisation and differential evolution (DE) algorithm was used for parameter optimisation. Furthermore, the generalization capability of BRBES was shown. This research work illustrates the efficiency of DE for parameter optimisation of BRBES. However, there is a lack of finding optimal values for the control parameters of DE, which may lead to more better results.

There have been several attempts to determine the near right values of the control parameters for DE [16], [17], [18]. Among different approaches, adaptive parameters control is one of the suitable mechanisms to determine the near right values of the control parameters for DE due to its simplicity. Liu et al. [19] proposed an adaptive mechanism to discover optimal values for F and CR for DE using the fuzzy logic controller (FLC), which is known as fuzzy adaptive differential evolution (FADE). However, the fuzzy-based systems cannot address all types of uncertainty except imprecision. Recently, Leon et al. [20] proposed a new adaptive algorithm based on the greedy algorithm, named greedy adaptive differential evolution (GADE). The algorithm uses the greedy search algorithm to determine the better parameter using neighbourhood points of a current population. The proposed work does not have any mechanism to address uncertainty. Tanabe et al. [21] proposed a new adaptive DE, named L-SHADE, where the control parameters are selected based on the success-history of F and CR. Furthermore, they introduced a mechanism for reducing the population size during each iteration. However, L-SHADE also lacked any mechanism to address the uncertainty of objective function.

In summary, the joint optimisation of parameter and structure for BRBES has shown better results among different optimal learning procedures mentioned above. Among evolutionary algorithms, DE is preferable for joint optimisation of BRBES for its better dealing with multiple local minima. However, there is a lack of determining the right values for the control parameters of DE while addressing the uncertainty of the objective function [22], [23]. Finding near right values of the control parameters and ensuring the balance of exploration and exploitation represents a challenge for using DE as an optimal training procedure for BRBES. Therefore, a new learning approach is required to improve the accuracy of BRBES.

III. Methodology

This section presents the DE as well as the BRBES, which can be used to address the uncertainty associated with the objective function of a problem domain like a power generation plant or a data centre. This is followed by the presentation of eBRBaDE, where the right amount of exploration and exploitation will be demonstrated necessary to make an algorithm balanced. Taking eBRBaDE into account, an integrated learning approach based on joint optimisation will be presented.

A. Differential Evolution

DE is preferable to support optimal training procedure for parameter optimisation due to its capability of avoiding local optima and finding global optimal. DE is a stochastic and population-based meta-heuristic algorithm. A population contains a set of individuals where each individual represents a possible solution to the problem. The population evolves from one iteration to another during the optimisation process. In Figure 1, it is shown that a population is initialized at the beginning. Then a new mutated or donor population is generated during the mutation step using mutation factor F. In the next step, the crossover is performed between the original and mutated population using crossover factor CR, which generates a trial population. Afterwards, a new population is generated by selecting the best individuals between original and trial population during the selection step. Finally, the stopping criteria are checked; if any criterion is fulfilled, then the individual of the new population with the best fitness value is considered as an optimal solution; otherwise, the process continues. A detailed description of the mutation, crossover, and selection steps are given below.

Here, $X_{i,g}$ denotes the i^{th} individual in the population at iteration g, where $i = 1, 2, \ldots, NP$, and NP denotes the population size. Each individual X_i has D dimensions, where $X_i = x_{i,1}, x_{i,2}, \ldots, x_{i,D}$. DE has three steps in every iteration, which are mutation, crossover, and selection. A population is initialized at the beginning.

During mutation a new mutated or donor population $V_{i,g}$ is generated based on the original population $X_{i,g}$. Eq. (1) is commonly used to generate a mutated population.

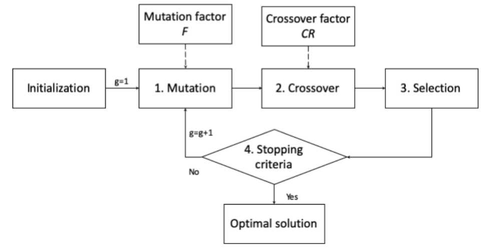

Fig. 1: The flowchart of DE.

$$V_{i,g} = X_{r_1,g} + F \times (X_{r_2,g} - X_{r_3,g}) \tag{1}$$

where r_1, r_2, r_3 are mutually exclusive and randomly selected from $X_{i,g}$. The mutation factor F is a real positive number from the interval [0, 2].

In the next step, which is known as crossover, a new trial population $T_{i,g}$ is generated from the mutation $V_{i,g}$ and original $X_{i,g}$ populations. Every individual of the trial population is derived using Eq. (2).

$$T_{i,g}[j] := \begin{cases} V_{i,g}[j] \text{ IF } rand[0,1] < CR \\ X_{i,g}[j] \text{ Otherwise} \end{cases} \tag{2}$$

where j represents an individual in a population, CR is the probability of recombination and $rand$ is a random value from the interval [0, 1] to ensure that at least one individual from the mutant vector is selected. The value of CR is also from the interval [0, 1] [19].

The last step is known as selection. In this step, each individual of the original $X_{i,g}$ and trial $T_{i,g}$ populations are compared based on fitness value generated by the objective function. The best individual of them is added to the original population for the next generation. The process of selecting individuals from original and trial populations is shown in Eq. (3).

$$X_{i,g+1}[j] := \begin{cases} T_{i,g} \text{ IF } f(T_{i,g}) < f(X_{i,g}) \\ X_{i,g} \text{ Otherwise} \end{cases} \tag{3}$$

After completion of these three steps a new generation of the population is generated. This process continues for a predefined number of iterations or stops when the optimisation function reaches a pre-set threshold value.

The objective function can contain uncertainty or noise. For example, Eq. (4) can be considered as an objective function. Here, z_m represents the predicted output value generated based on various inputs and $\bar{z}_m$ is the actual output value. The inputs can be erroneous or misleading due to unpredicted circumstances which have not been considered while designing the system. The impact of this noise or uncertainty can be addressed using BRBES. Henceforth, a detailed description of addressing uncertainty by BRBES, is presented in the next subsection.

$$\xi_{(p)} = \frac{1}{M} \sum_{m=1}^{M} (z_m - \bar{z}_m)^2 \tag{4}$$

B. Belief Rule-Based Expert System (BRBES)

In this subsection, the process of calculating the value of z_m from Eq. (4) using BRBES and the process of addressing various uncertainty will be described. The output z_m is usually calculated from a fuzzy value where the degree of belief is associated. The output is produced from the input using the inference procedure. Based on the number of input variables belief rules are generated. A belief rule has two parts: one is antecedent or premise part, which consists of antecedent attributes; while the other is consequent or conclusion part which contains the consequent attribute. The antecedent attributes use referential values, and the belief degrees are associated with the consequent attribute of the belief rule, which is shown in Eq. (5). Each belief rule is assigned with a rule weight to show its importance.

$$R_k : \begin{cases} \text{IF } (A_1 \text{ is } V_1^k) \text{ AND / OR } (A_2 \text{ is } V_2^k) \text{ AND / OR} \\ \ldots \text{ AND / OR } (A_{T_k} \text{ is } V_{T_k}^k) \\ \text{THEN } (C_1, \beta_{1k}), (C_2, \beta_{2k}), \ldots, (C_N, \beta_{Nk}) \end{cases} \tag{5}$$

$$\text{where } \beta_{jk} \geq 0, \sum_{j=1}^{N} \beta_{jk} \leq 1 \text{ with rule weight } \theta_k,$$

$$\text{and attribute weights } \delta_{k1}, \delta_{k2}, \ldots \delta_{kTk}, k \in 1, \ldots, L$$

where $A_1, A_2, \ldots, A_{T_k}$ are the antecedent attributes of the k^{th} rule. $V_i^k (i = 1, \ldots, T_k,\ k = 1, \ldots, L)$ is the referential value of the i^{th} antecedent attribute. C_j is the j^{th} referential value of the consequent attribute. $\beta_{jk} (j = 1, ..., N, k = 1, ..., L)$ is the degree of belief for the consequent reference value C_j. If $\sum_{j=1}^{N} \beta_{jk} \leq 1$, then the k^{th} rule is considered as complete; otherwise, it is incomplete.

Usually, the set of belief rules constitutes the Belief Rule Base (BRB). The logical connectives of the antecedent attributes in a belief rule can be either AND or OR, which represents the conjunctive or the disjunctive assumptions of the rule, respectively. Based on the logical connectivity of the BRB, a BRBES can be named as conjunctive or disjunctive BRBES.

After constructing the BRB, the inference procedure is used to generate the output. The inference procedure consists of four steps; input transformation, rule activation, belief update, and rule aggregation using the evidential reasoning approach. The input data is distributed over the referential values of the antecedent attributes, which is called the matching degree during the input transformation. Afterwards, the belief rules are called packet antecedent. Subsequently, activation weights of the rules are calculated using matching degrees.

The activation weight w_k for the k^{th} rule for the conjunctive assumption is calculated using the following expression:

$$w_k = \frac{\theta_k \prod_{i=1}^{T_k} \alpha_i^k}{\sum_{i=1}^{L} (\theta_k \prod_{i=1}^{T_k} \alpha_i^l)} \tag{6}$$

Here, θ_k is the rule weight and α_k is the matching degree of the k^{th} rule. As, in the conjunctive assumption all matching degrees are multiplied.

However, for the disjunctive assumption the activation weight w_k for the k^{th} rule is calculated using the following expression:

$$w_k = \frac{\theta_k \sum_{i=1}^{T_k} \alpha_i^k}{\sum_{i=1}^{L} (\theta_k \sum_{i=1}^{T_k} \alpha_i^l)} \tag{7}$$

Here, θ_k is the rule weight and α_k is the matching degree of the k^{th} rule. In the disjunctive assumption all matching degrees are summed.

If any of the antecedent attributes are ignored, the belief degree associated with each belief rule needs to be updated. The belief degree update is generated using the method presented in [1]. Afterwards, the rule aggregation is performed using the recursive reasoning algorithm as shown in Eq. (8) [24]

$$\beta_j = \frac{\mu \times [X - \prod_{k=1}^{L} (1 - \omega_k \sum_{j=1}^{N} \beta_{jk})]}{1 - \mu \times [\prod_{k=1}^{L} 1 - \omega_k]} \tag{8}$$

$$\text{where } X = \textstyle\prod_{k=1}^{L} (\omega_k \beta_{jk} + 1 - \omega_k \sum_{j=1}^{N} \beta_{jk})$$

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

Here, ω_k is the activation weight of the k^{th} rule, while β_{jk} denotes the belief degree related to one of the consequent reference values.

The fuzzy output of rule aggregation procedure is converted to a crisp value using the utility values of the consequent attribute, which is considered as the final result, as shown in Eq. (9).

$$z_i = \sum_{j=1}^{N} u(O_j) \beta_j \tag{9}$$

where z_i is the expected numerical value and $u(Oj)$ is the utility score of each referential value.

C. Enhanced Belief Rule-Based Adaptive Differential Evolution (eBRBaDE)

BRBES's capability of addressing various uncertainty can be incorporated with DE to address the uncertainty of the objective function. Furthermore, DE is highly influenced by mutation and crossover factors. The mutation (F) and crossover (CR) factors can be adapted to improve DE performance. It is evident that F and CR may change during each iteration of DE, which facilitates in finding optimal values more efficiently. Most of the research works [19], [20] of DE have considered changing the parameter values based on fitness values of the objective function. However, they did not consider the different types of uncertainty related to DE approaches. Therefore, we propose a BRBES based DE parameter adaptation algorithm, named eBRBaDE, which is able to deal with all kinds of uncertainty due to the BRBES's inherent capability of addressing various uncertainty. Figure 2 depicts the system diagram of eBRBaDE.

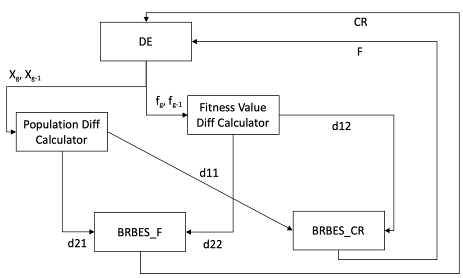

Fig. 2: Enhanced belief rule-based adaptive differential evolution (eBRBaDE).

In general, changes of the population (PC) and objective function (FC) values of each individual in each iteration is passed to two BRBESs, named BRBES_F and BRBES_CR, as input as illustrated in Figure 2. Afterwards, based on the belief rule base and using the evidential reasoning approach, new F and CR values are computed by BRBES_F and BRBES_CR respectively for next iteration of each individual of the population as shown in Figure 2. The BRBES helps to achieve the right amount of exploration and exploitation of the search space by determining F and CR values based on the changes of the population and the objective function values in each iteration. Using the new F and CR values the mutation, crossover, and selection steps of DE are performed, and the new population is generated. This process continues until the stop criteria are fulfilled, and the individual of the new population with the best fitness value is selected as an optimal solution.

$$PC_i = \sum_{j=1}^{D}(x_{j,i}^{g} - x_{j,i}^{(g-1)})^2; i = 1, ..., NP; j = 1, ..D \quad (10)$$

$$FC_i = (f_i^{g} - f_i^{(g-1)})^2; i = 1, ..., NP; j = 1, ..D \quad (11)$$

$$mmPC_i = movemean(PC, tr) \quad (12)$$

$$mmFC_i = movemean(FC, tr) \quad (13)$$

$$d11_i = 1 - (1 + mmPC_i)e^{-mmPC_i} \quad (14)$$

$$d12_i = 1 - (1 + mmFC_i)e^{-mmFC_i} \quad (15)$$

$$d21_i = 2(d11_i) \quad (16)$$

$$d22_i = 2(d12_i) \quad (17)$$

Here PC is the change in individual of the population between two iterations. $x_{j,i}^{g}$ and $x_{j,i}^{g-1}$is the individual of the population on the g^{th} and $(g-1)^{\text{th}}$ iteration respectively. FC is the change in objective function between two iterations, while the f_i^g and $f_i^{(g-1)}$ are the function values for the i^{th} population on g^{th} and $(g-1)^{\text{th}}$ iteration respectively. $mmPC_i$ and $mmFC_i$ is the mean value of a sliding window of tr length of PC_i and FC_i. $movemean$ is a Matlab function to calculate sliding window average.

The values of PC_i and FC_i are calculated using Eqs. (10) and (11). Afterwards, a sliding window average of the value of PC_i and FC_i were computed with threshold of tr as shown in Eqs. (12) and (13). The threshold value tr was chosen based on empirical analysis. The sliding window average helps to reduce abrupt changes in magnitude in accordance with previous values. The value of $mmPC_i$ and $mmFC_i$ have been rescaled between 0 and 1 using Eqs. (14) and (15) to fit the preferred value of CR. $d11_i$ and $d12_i$ are later used as input for the BRBES_CR for predicting the new values of CR. Similarly, using Eqs. (16) and (17) the value of $mmPC_i$ and $mmFC_i$ have been rescaled between 0 and 2 to fit the preferred value of F. For predicting the values of F BRBES_F is used where $d21_i$ and $d22_i$ are used as input.

The algorithm is presented as pseudocode in Algorithm 1, where D denotes the dimension of individual, F the mutation factor, CR the crossover factor, NP the population size, X_U_i and X_L_i the upper and lower bounds of the i^{th} individual. $f(X_{i,g})$ represents the objective function which is to be optimised, while $best\_solution$ represents the optimal solution. After the second iteration, using Eqs. (10), (11), (14), (15), (16), and (17), the $d11$, $d12$, $d21$ and $d22$ values are obtained, which are later passed on as inputs to the BRBES_CR and BRBES_F as shown in line number 10 and 11 of Algorithm 1 to get the new values of F and CR. Then mutation and crossover steps are performed in line numbers 14 to 18 using the new values of F and CR. A new trial population $T_{i,g}$ is generated after mutation and crossover steps. Among $T_{i,g}$ and $X_{i,g}$ the individuals with best fitness value is selected and assigned to new population $X_{i,g+1}$ as mentioned in line numbers 19 to 20. The new population $X_{i,g+1}$ which will be used for g+1 iteration. This loop will continue until $MAX\_Iteration$ is reached or $best\_value$ does not change for $S1$ iterations. Afterwards, the individual with the lowest fitness among the population is returned as optimal solution.

Two different BRBESs, named BRBES_CR and BRBES_F, are used for calculating the values of CR and F respectively, as shown in Figure 3. BRBES_CR consists of two antecedents, namely d11 and d12 while CR being its consequent. Similarly,

Algorithm 1 eBRBaDE algorithm

Let D denote the dimension, G the generation, F the mutation factor, CR the crossover factor, NP the population size, $X_{i,g}$ the i_{th} individual in the population of G_{th} generation, X_U_i and X_L_i upper and lower bounds of the i_{th} individual.

Input D, F, CR, NP, X_U and X_L

Output Optimised values: $best\_solution$

procedure eBRBaDE(D, F, CR, NP, X_U, X_L)
  Initialize population, $P := X_{1,0}, X_{2,0}, \cdots, X_{NP,0}$
  for each i $\in$ P **do**
    $X_{i,0} := X\_L_{i,0} + rand[i, 0] * (X\_U_{i,0} - X\_L_{i,0})$
  $G := 1$
  $Max\_Iteration := 3000$
  while $G <= MAX\_Iteration$ AND No change in $best\_value$ for $S1$ iterations **do**
    if $G > 2$ **then**
      for all $i <= NP$ **do**
        Get $d11$, $d12_i$, $d21_i$, and $d22_i$ values using Eqs. (10), (11), (14), (15), (16), and (17).
        Get the predicted values of f_i and cr_i using BRBES_F and BRBES_CR.
      $F := mean(f)$
      $CR := mean(cr)$
    for all $i <= NP$ **do**
      Randomly select r_1, r_2, $r_3 \in (1, \cdots, NP)$
      for all $j <= D \wedge j \notin (r_1, r_2, r_3)$ **do**
        $V_{j,g} := X_{r_1,g} + F \times (X_{r_2,g} - X_{r_3,g})$
        $$T_{j,g}[j] := \begin{cases} V_{i,g}[j] \text{ IF } rand[0,1] < CR \\ X_{i,g}[j] \text{ Otherwise} \end{cases}$$
    for all $i <= NP$ **do**
      $$X_{i,g+1} := \begin{cases} T_{i,g} \text{ IF } f(T_{i,g}) + \text{PENALTY}(T_{i,g}) \\ < f(X_{i,g}) + \text{PENALTY}(X_{i,g}) \\ X_{i,g} \text{ Otherwise} \end{cases}$$
    $G := G + 1$
  $best\_value := min(f(X_{g+1}))$
  $best\_solution :=$ Select $X_{i,g+1} \in X_{g+1}$ which generates the lowest $best\_value$
  return $best\_solution$

BRBES_F also has two antecedents; d21 and d22 while F being the consequent attribute. The utility values of the antecedent attributes (d11, d12) and consequent attribute (CR) of BRBES_CR are set between 0 and 1 as it is the preferred value for CR [25]. The utility values of the antecedent attribute (d21, d22) and consequent attribute (F) of BRBES_F are set between 0 and 2 as it is the preferred value for F [25]. The range of F and CR values were equally divided among the utility values. From Table I it can be observed that BRBES_CR and BRBES_F are calculating F and CR with higher accuracy for six referential values of the antecedent and consequent attributes.

TABLE I: Comparison of MSE among various referential values of BRBES_CR and BRBES_F

Number of referential values		MSE
d11 or d21	d12 or d22	
3	3	4.15E-04
5	5	3.16E-04
6	6	4.66E-05
7	7	5.23E-05

Tables II and III present the final referential and utility values for BRBES_CR and BRBES_F respectively. For BRBES_F, $d21$ and $d22$ are used as input, while $d11$ and $d12$ are used as input for BRBES_CR. $d11$ and $d12$ each have six referential values. The rule base helps BRBES to predict higher values of F and CR based on the inputs. Higher values of F and CR ensures more exploration of the search space, while lower values of F and CR helps on exploitation more. Therefore, F and CR values need to be adjusted based on the status of the optimisation process, which can be determined based on the values of PC and FC. A higher FC value indicates that the solution is far away from convergence and vice versa. On the other hand, a higher PC value indicates the individuals of the populations are highly distributed over the search space and vice versa. For example, if FC has a higher predicted value, then it means that the current population is far from an optimal solution. Therefore, by increasing the values of F and CR the exploration of the search space can be ensured, which in turn allow eBRBaDE to find a near-optimal solution.

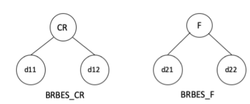

Fig. 3: Two BRBESs used for eBRBaDE.

TABLE II: Details of BRBES_CR

	Antecedent Attributes						Consequent Attribute					
	d11, d12						CR					
Referential Values	H	HM	M	ML	LL	L	H	HM	M	ML	LL	L
Utility Values	1	0.8	0.6	0.4	0.2	0	1	0.8	0.4	0.2	0.1	0.1

TABLE III: Details of BRBES_F

	Antecedent Attributes						Consequent Attribute					
	d21, d22						F					
Referential Values	H	HM	M	ML	LL	L	H	HM	M	ML	LL	L
Utility Values	2	1.6	1.2	0.8	0.4	0	2	1.6	1.2	0.8	0.4	0.1

The proposed eBRBaDE procedure provides a solution for dealing with the uncertainty in the objective function by incorporating BRBES with DE. Furthermore, BRBES helps in predicting the values of F and CR based on the changes in population and fitness values between two consecutive iterations. The anticipated F and CR values control the mutation and crossover of DE, which generates the new population, which in turn navigates the traversal of the search space. The belief rules control the predicted values of F and CR to ensure the right amount of exploration and exploitation of the search space by DE. In the next subsection, a new learning approach is presented for BRBES using eBRBaDE as an optimal training procedure.

D. Learning in BRBES based on eBRBaDE

This subsection describes the process of incorporating learning with BRBES using an optimal training procedure, eBRBaDE. Different parameters of BRBES, like attribute weights (θ_k), rule weights (δ_i), and belief degrees (β_k) play an important role in the accuracy of the results. These parameters are usually known as learning parameters, which are generally assigned by domain experts, or they are randomly selected. The antecedent attributes and belief rules are prioritized using the attribute weights and rule weights consecutively. Belief degrees of the consequent attribute is used to present the uncertainty of the output. Hence, the learning parameters are essential for BRBES. Therefore, a suitable method is needed to find the optimal values of the learning parameters. By training the BRBES with data, the optimal values of the learning parameters can be discovered [7].

The learning parameters need to be trained to determine the optimal values by an objective function considering the linear equality and inequality constraints. The output from BRBES is considered as simulated output (z_m) and output from the system is named observed output ($\bar{z}_m$). The difference $\xi(p)$ between simulated and observed output needs to be minimized by the optimization process The training sample contains M data points, where the input for BRBES is u_m, the observed output is $\bar{z}_m$, and the simulated output is z_m ($m = 1, \ldots, M$). The error $\xi(p)$ is measured by Eq. (4).

The optimisation of the learning parameters is conducted as defined in the following equation:

$$\min_{P} \xi(p) \tag{18}$$

$$P = P(\mu(O_j), \theta_k, \delta_k, \beta_{jk})$$

The objective function for training the BRBES consists of Eqs. (8) and (9). Furthermore, to ensure the completeness of the belief rules, the summations of the belief degree for each rule should be one. Additionally, the values of attribute weights, rule weights, and belief degrees are considered between zero and one. Henceforth, to enforce the above-mentioned criteria the following constraints are considered:

- Utility values of the consequent attributes $\mu(O_j)(j = 1, \ldots, n)$:
  $\mu(O_i) < \mu(O_j);\text{If } i < j$
- Rule weights $\theta_k(k = 1, \ldots, K)$:
  $1 \geq \theta_k \geq 0;$
- Antecedent attribute weights $\delta_k, (k = 1, \ldots, K)$:
  $1 \geq \delta_k \geq 0;$
- Consequent belief degrees for the kth rule $\beta_{jk}, (j = 1, \ldots, n, k = 1, \ldots, L)$:
  $1 \geq \beta_{jk} \geq 0;$
  $\sum_{j=1}^{n} \beta_{jk} = 1;$

Yang et. al. [9] have proven the benefits of parameter and structure optimisation for BRBES to generate more accurate results. Therefore, eBRBaDE as described in Subsection III-C, will be used for parameter optimisation of BRBES. However, to enforce the constraints of BRBES, Eq. (3) has been modified and a penalty function has been added as shown in Eq. (19). By using this mechanism, it can be ensured that individuals of the populations which do not satisfy the constraints will have lower fitness values.

$$X_{i,g+1}[j] := \begin{cases} T_{i,g} \text{ IF } f(T_{i,g}) + penalty(T_{i,g}) < f(X_{i,g}) \\ +\text{PENALTY}(X_{i,g}) \\ X_{i,g} \text{ Otherwise} \end{cases} \tag{19}$$

In the beginning, all the learning parameters and the structure of the BRBES need to be initialised. Afterwards, the parameter optimisation is carried out using the eBRBaDE algorithm. The eBRBaDE optimal training procedure generates the optimal values of the learning parameters for the initial BRB structure of BRBES. The initial BRB is updated with the optimised values of the learning parameters. Afterwards, the stop criterion is checked. For the first iteration, it will be false and the process moves to the structure optimisation step. The structure optimisation of the initial BRB is performed using the Heuristic Strategy (SOHS) algorithm, as mentioned in [9]. Then parameter optimisation is performed using the learning parameters of the new structure. These iterations continue until the structure does not change for S_2 iterations. The above described eBRBaDE based joint optimisation learning mechanism is presented in Figure 4.

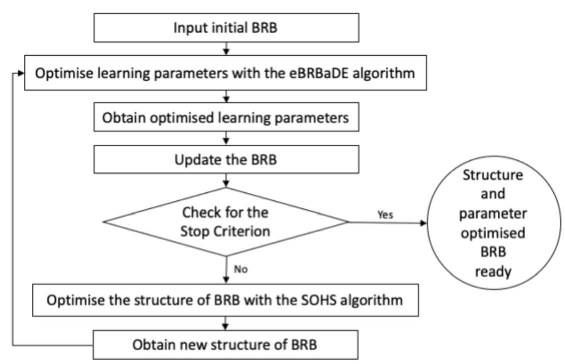

Fig. 4: Flowchart of eBRBaDE based learning mechanism

Therefore, a new optimal training procedure has been proposed as a learning approach for BRBES by using a joint optimisation framework of eBRBaDE as parameter and SOHS as structure optimisation technique.

IV. Results

In this section, the performance of the proposed new learning mechanism is evaluated in details. Evaluation techniques play a significant role in measuring the performance of a learning mechanism. Researchers proposed various performance measurements metrics. Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) are some of the standard techniques used for comparing the performance among different learning algorithms [26]. The metrics, as mentioned above, are used to evaluate the performance of the proposed new learning mechanism with others. The newly proposed eBRBaDE based learning mechanism has been compared with various evolutionary algorithms such as BAT, PSO, DE, and L-SHADE. In this study, eBRBaDE and other evolutionary algorithms are used to incorporate learning with BRBES for predicting the power usage effectiveness (PUE) of a datacentre in UK. The above-mentioned performance measurement metrics have been used to compare the accuracy of the PUE prediction among different learning mechanisms. Furthermore, to investigate the impact of our proposed learning mechanism in details, both the disjunctive and conjunctive BRBES have been used for predicting the PUE of a datacentre.

A. Use Case Scenario

We collected a dataset from the datacentre of Leeds Beckett University for predicting PUE [24]. The dataset contained data of the following fields; external temperature, server room temperature, IT equipment energy usage, and PUE with a sample rate of 30 minutes. After pre-processing and removing the data containing erroneous measurement from the dataset, a total of 5,300 data points was taken from the dataset for predicting PUE. The dataset was divided into 80:20 ratio for training and testing the learning algorithm. All experiments were conducted on a MacBook Pro with Intel Core i7 processor, 2.2 GHz, and 16 GB RAM.

B. Initial Belief Rule Base

Based on the dataset, an initial BRB was developed, where external temperature, server room temperature, and IT equipment energy usage were considered as the antecedent attributes, while the PUE value was considered as the consequent attribute, which is illustrated in Figure 5. PUE, External temperature, server room temperature, and IT equipment energy usage are refereed as “X1”, “X2”, “X3”, and “X4” respectively.

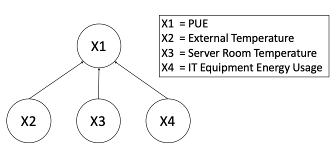

Fig. 5: BRB tree for PUE

Table IV presents the referential and utility values of the antecedent and consequent attributes. Each of the attributes have been assigned three referential values which are “H”, “M”, and “L”. The maximum and minimum value of each of the attributes in the dataset are considered as the utility values for “H” and“L”. The average of maximum and minimum of the attributes in the dataset is assigned as the utility value for the referential value “M”.

C. Comparison of learning mechanisms for disjunctive BRBES

The eBRBaDE based learning mechanism is used to incorporate learning for the disjunctive BRBES. The performance of eBRBaDE is compared with other evolutionary algorithms, such as BAT, PSO, GA, DE, and L-SHADE [13], [21]. In this comparison, the evolutionary algorithms are used for parameter optimisation, and the SOHS algorithm [9] is used for structure optimisation for predicting the PUE values. We have used k-fold cross-validation for our experiments. Usually, 5 or 10 folds are commonly used for cross-validation. Considering the execution time fivefold cross-validation is used in these experiments [26]. The Mean Squared Error (MSE) is used to measure the accuracy of eBRBaDE based learning mechanism.

TABLE IV: Initial structure of the BRB for antecedent and consequent part

	Antecedent Attributes									Consequent Attribute		
	X2			X3			X4			X1		
Attribute Weights	1			1			1					
Referential Values	H	M	L	H	M	L	H	M	L	H	M	L
Utility Values	12.20	7.95	3.70	26.00	23.50	21.00	434947.97	283010.50	131073.03	0.45	0.35	0.25

The MSE values of PUE prediction for different learning mechanisms during training and testing is presented in Tables V and VI as well as in Figure 6. Table V represents the MSE values of each fold for BAT, PSO, GA, DE, L-SHADE, and eBRBaDE during training. The last three rows represent the average, standard deviation, and the minimum values of the fivefolds for each of the algorithms. Among the evolutionary algorithms, eBRBaDE has the lowest average and minimum MSE values, which are 1.58E-06 and 3.71E-07 respectively.

TABLE V: MSE values of training datasets for BAT, PSO, GA, DE, L-SHADE and eBRBaDE for disjunctive BRBES

	BAT	PSO	GA	DE	L-SHADE	eBRBaDE
1st Fold	1.30E-04	1.21E-04	6.52E-05	3.00E-05	2.60E-06	4.51E-07
2nd Fold	1.32E-04	1.15E-04	6.03E-05	2.18E-05	1.21E-06	1.49E-06
3rd Fold	1.29E-04	1.20E-04	7.88E-05	2.52E-05	1.75E-06	3.12E-06
4th Fold	1.20E-04	1.16E-04	4.60E-05	3.53E-05	3.61E-06	3.71E-07
5th Fold	1.31E-04	1.02E-04	7.86E-05	4.07E-05	3.91E-06	2.47E-06
Average	1.28E-04	1.15E-04	6.58E-05	3.06E-05	2.62E-06	1.58E-06
Standard Deviation	4.67E-06	7.61E-06	1.37E-05	7.59E-06	1.16E-06	1.22E-06
Minimum	1.20E-04	1.02E-04	4.60E-05	2.18E-05	1.21E-06	3.71E-07

The MSE values of each fold for BAT, PSO, GA, DE, L-SHADE, and eBRBaDE during test are presented at Table VI. The last three rows of Table VI represents the average, standard deviation and the minimum MSE values of the fivefolds for each of the algorithms. The eBRBaDE has the lowest average and minimum MSE among the others, which are 2.60E-06 and 5.50E-07. It can also be observed that the average and minimum MSE values of testing are comparatively higher than that of testing for eBRBaDE, which implies that the models are not over-fitted.

TABLE VI: MSE values of testing datasets by BAT, PSO, GA, DE, L-SHADE and eBRBaDE for disjunctive BRBES

	BAT	PSO	GA	DE	L-SHADE	eBRBaDE
1st Fold	1.28E-04	1.21E-04	6.48E-05	2.43E-05	1.68E-06	5.50E-07
2nd Fold	1.36E-04	1.21E-04	7.41E-05	1.38E-05	3.38E-06	3.34E-06
3rd Fold	1.07E-04	1.03E-04	5.70E-05	1.21E-05	1.21E-06	2.54E-06
4th Fold	1.66E-04	1.63E-04	9.30E-05	6.96E-05	7.84E-06	4.90E-06
5th Fold	1.01E-04	6.86E-05	4.54E-05	3.06E-05	4.27E-06	1.67E-06
Average	1.27E-04	1.15E-04	6.68E-05	3.01E-05	3.68E-06	2.60E-06
Standard Deviation	2.60E-05	3.41E-05	1.80E-05	2.33E-05	2.64E-06	1.65E-06
Minimum	1.01E-04	6.86E-05	4.54E-05	1.21E-05	1.21E-06	5.50E-07

The average MSE values of fivefold cross-validation for

BAT, PSO, GA, DE, L-SHADE, and eBRBaDE during training and testing are illustrated in Figure 6. From this figure, it can be concluded that the eBRBaDE is performing better than the others during testing and training.

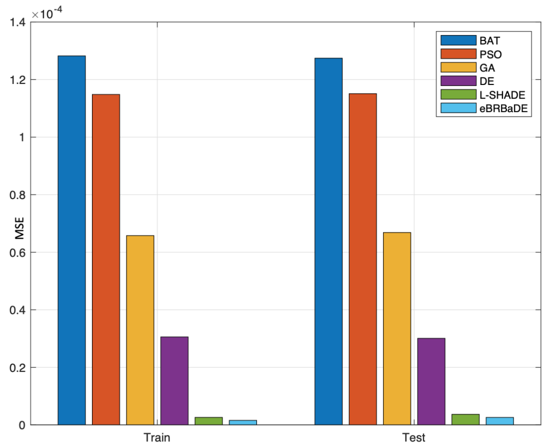

Fig. 6: Comparison of MSE for disjunctive BRBES with PSO, GA, BAT, DE, L-SHADE, and eBRBaDE

To further evaluate the performance of eBRBaDE RMSE, MAE, and MAPE is used for the predicted PUE values based on the test dataset, which is presented in Table VII. From the table it can be observed that disjunctive BREBES which is trained by eBRBaDE is performing better than BAT, PSO, GA, DE, and L-SHADE.

TABLE VII: Comparison of RMSE, MAE, and MAPE values of predicted PUE for testing datasets by BAT, PSO, GA, DE, L-SHADE and eBRBaDE for Disjunctive BRBES

	BAT	PSO	GA	DE	L-SHADE	eBRBaDE
RMSE	1.13E-02	1.10E-02	8.78E-03	4.93E-03	7.42E-04	7.25E-04
MAE	9.22E-03	8.53E-03	5.76E-03	2.07E-03	2.08E-04	2.07E-04
MAPE	2.10E+00	1.96E+00	1.31E+00	4.88E-01	4.83E-02	4.81E-02

D. Comparison of learning mechanisms for conjunctive BRBES

To further evaluate the performance of eBRBaDE based learning mechanism, it has been used to train a conjunctive BRBES for predicting PUE values for the same dataset. The MSE values of each fold for BAT, PSO, GA, DE, L-SHADE, and eBRBaDE during training is presented in Table VIII. The last three rows of Table VIII represent the average, standard deviation, and the minimum values of the fivefolds for each of the algorithms. The L-SHADE has the lowest average and minimum MSE among the others, which are 6.05E-08 and 1.73E-08.

The MSE values of each fold for BAT, PSO, GA, DE, L-SHADE, and eBRBaDE during testing is presented in Table IX. The average, standard deviation, and the minimum values of the fivefolds for each of the algorithms are presented in the last three rows of Table VI. The eBRBaDE has the lowest average and minimum MSE among all algorithms, which are 7.72E-06 and 1.65E-08, even though it did not

TABLE VIII: MSE values of training datasets by BAT, PSO, GA, DE, L-SHADE and eBRBaDE for conjunctive BRBES

	BAT	PSO	GA	DE	L-SHADE	eBRBaDE
1st Fold	1.50E-04	1.26E-04	7.78E-05	3.26E-05	9.44E-08	2.16E-07
2nd Fold	3.14E-04	1.25E-04	9.18E-05	3.55E-05	7.90E-08	4.64E-08
3rd Fold	1.47E-04	1.49E-04	1.00E-04	4.17E-05	8.50E-08	1.47E-07
4th Fold	1.17E-04	1.15E-04	8.10E-05	2.92E-05	1.73E-08	4.60E-08
5th Fold	2.69E-04	1.32E-04	9.40E-05	3.78E-05	2.67E-08	1.95E-08
Average	2.00E-04	1.29E-04	8.90E-05	3.54E-05	6.05E-08	9.49E-08
Standard Deviation	8.65E-05	1.25E-05	9.40E-06	4.80E-06	3.57E-08	8.32E-08
Minimum	1.17E-04	1.15E-04	7.78E-05	2.92E-05	1.73E-08	1.95E-08

TABLE IX: MSE values of test datasets by BAT, PSO, GA, DE, L-SHADE and eBRBaDE for conjunctive BRBES

	BAT	PSO	GA	DE	L-SHADE	eBRBaDE
1st Fold	1.52E-04	1.25E-04	7.62E-05	2.00E-05	9.13E-08	2.38E-07
2nd Fold	3.23E-04	1.29E-04	1.02E-04	3.18E-05	1.20E-07	5.87E-08
3rd Fold	1.22E-04	1.29E-04	7.63E-05	1.41E-05	7.28E-08	1.48E-07
4th Fold	1.68E-04	1.58E-04	1.28E-04	6.17E-05	4.05E-05	3.81E-05
5th Fold	2.41E-04	1.03E-04	6.57E-05	4.65E-06	2.77E-08	1.65E-08
Average	2.01E-04	1.29E-04	8.98E-05	2.65E-05	8.17E-06	7.72E-06
Standard Deviation	8.11E-05	1.97E-05	2.55E-05	2.21E-05	1.81E-05	1.70E-05
Minimum	1.22E-04	1.03E-04	6.57E-05	4.65E-06	2.77E-08	1.65E-08

have the lowest MSE values during training. It can also be observed that the average and minimum MSE values of testing are comparatively higher than that of training for eBRBaDE, which suggests that the models are not over-fitted.

Figure 7 illustrates the average MSE values of fivefold cross-validation for BAT, PSO, GA, DE, L-SHADE, and eBRBaDE during training and testing. It can be observed that the eBRBaDE is performing better than the other algorithms during testing.

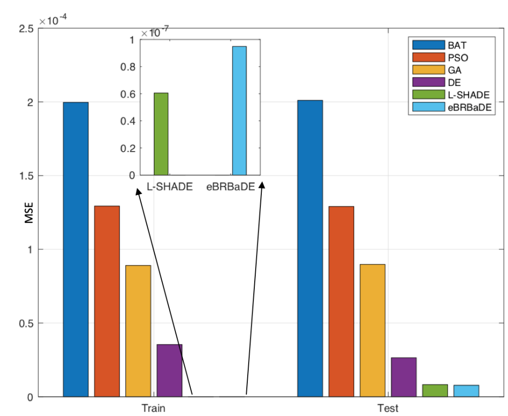

Fig. 7: Comparison of MSE for conjunctive BRBES with PSO, GA, BAT, DE, and eBRBaDE

To further evaluate the performance of eBRBaDE, RMSE, MAE, and MAPE values are calculated for the predicted PUE values based on the test dataset, which is presented in Table X. From the table it can be observed that Conjunctive BREBES trained by eBRBaDE is performing significantly better than the BAT, PSO, GA, DE, and L-SHADE.

In summary, our proposed joint optimisation learning mechanism based eBRBaDE has been used to predict PUE values of a datacentre using conjunctive and disjunctive BRBES. The results are shown in Subsections IV-C and IV-D for disjunctive and conjunctive BRBES respectively. From the results, it can be concluded that trained conjunctive BRBES predicated PUE

TABLE X: Comparison of RMSE, MAE, and MAPE of predicted PUE for testing datasets by BAT, PSO, GA, DE, L-SHADE and eBRBaDE for conjunctive BRBES

	BAT	PSO	GA	DE	L-SHADE	eBRBaDE
RMSE	1.55E-02	1.01E-02	8.10E-03	2.16E-03	1.66E-04	1.29E-04
MAE	1.34E-02	8.94E-03	6.60E-03	1.49E-03	8.84E-05	7.34E-05
MAPE	3.10E+00	2.02E+00	1.48E+00	3.37E-01	1.98E-02	1.65E-02

values with higher accuracy than trained disjunctive BRBES. The eBRBaDE based optimal training procedure has been compared with various evolutionary algorithms, such as BAT, PSO, GA, DE, and L-SHADE using different metrics. After analysing the results, it can be observed that eBRBaDE based learning mechanism is predicting PUE with higher accuracy in respect of MSE, RMSE, MAE, and MAPE in comparison with BAT, PSO, GA, DE, and L-SHADE. The eBRBaDe performed better than the other evolutionatry algorithm due to its capability of balanced exploration and exploitation of the search space.

V. Conclusion

In this study, a new optimal training procedure, named eBRBaDE, is proposed, by incorporating BRBES with DE for addressing the uncertainty of the objective function. The BRBES also helps to calculate the near-optimal values for the control parameters of DE, while ensuring balanced exploration and exploitation of the search space. A new learning mechanism for BRBES has been proposed so that it can produce results with higher accuracy. The new learning mechanism is based on joint optimisation using eBRBaDE as parameter and SOHS as structure optimisation procedures. The new eBRBaDE based learning mechanism helps to generate trained BRBES with more accurate results, as the eBRBaDE is able to maintain a balanced exploration and exploitation of the search space. The new learning approach has been compared with other optimal training procedures (like BAT, PSO, GA, DE, and L-SHADE) for predicting the PUE of a datacentre using the conjunctive and disjunctive BRBES. In this comparison, it has been found that trained conjunctive BRBES with joint optimisation using eBRBaDE predicted PUE leads to the lowest MSE. As future work, we intend to apply the proposed optimal training procedure and learning mechanism to different domains and more complex datasets to verify its feasibility further.

References

[1] J.-B. Yang, J. Liu, J. Wang, H.-S. Sii, and H.-W. Wang, "Belief rule-base inference methodology using the evidential reasoning approach-rimer," *IEEE Transactions on systems, Man, and Cybernetics-part A: Systems and Humans*, vol. 36, no. 2, pp. 266–285, 2006.

[2] A. A. Monrat, R. U. Islam, M. S. Hossain, and K. Andersson, "Challenges and opportunities of using big data for assessing flood risks," in *Applications of Big Data Analytics*. Springer, 2018, pp. 31–42.

[3] M. S. Hossain, F. Ahmed, and K. Andersson, "A belief rule based expert system to assess tuberculosis under uncertainty," *Journal of Medical Systems*, vol. 41, no. 3, pp. 1–11, 2017.

[4] M. S. Hossain, S. Rahaman, R. Mustafa, and K. Andersson, "A belief rule-based expert system to assess suspicion of acute coronary syndrome (ACS) under uncertainty," *Soft Computing*, vol. 22, pp. 1623–1639, 2018.

[5] M. S. Hossain, P.-O. Zander, M. S. Kamal, and L. Chowdhury, "Belief-rule-based expert systems for evaluation of e-government: a case study," *Expert Systems*, vol. 32, no. 5, pp. 563–577, 2015.

[6] L. Chang, Z. Zhou, Y. You, L. Yang, and Z. Zhou, "Belief rule based expert system for classification problems with new rule activation and weight calculation procedures," *Information Sciences*, vol. 336, pp. 75–91, 2016.

[7] J.-B. Yang, J. Liu, D.-L. Xu, J. Wang, and H. Wang, "Optimization models for training belief-rule-based systems," *IEEE Transactions on systems, Man, and Cybernetics-part A: Systems and Humans*, vol. 37, no. 4, pp. 569–585, 2007.

[8] L. Chang, J. Sun, J. Jiang, and M. Li, "Parameter learning for the belief rule base system in the residual life probability prediction of metalized film capacitor," *Knowledge-Based Systems*, vol. 73, pp. 69–80, 2015.

[9] L.-H. Yang, Y.-M. Wang, J. Liu, and L. Martínez, "A joint optimization method on parameter and structure for belief-rule-based systems," *Knowledge-Based Systems*, vol. 142, pp. 220–240, 2018.

[10] L.-L. Chang, Z.-J. Zhou, Y.-W. Chen, T.-J. Liao, Y. Hu, and L.-H. Yang, "Belief rule base structure and parameter joint optimization under disjunctive assumption for nonlinear complex system modeling," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 9, pp. 1542–1554, 2018.

[11] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Computing Surveys (CSUR)*, vol. 35, no. 3, pp. 268–308, 2003.

[12] R. D. Al-Dabbagh, F. Neri, N. Idris, and M. S. Baba, "Algorithmic design issues in adaptive differential evolution schemes: Review and taxonomy," *Swarm and Evolutionary Computation*, 2018.

[13] X.-S. Yang, *Nature-inspired optimization algorithms*. Elsevier, 2014.

[14] Z.-G. Zhou, F. Liu, L.-C. Jiao, Z.-J. Zhou, J.-B. Yang, M.-G. Gong, and X.-P. Zhang, "A bi-level belief rule based decision support system for diagnosis of lymph node metastasis in gastric cancer," *Knowledge-Based Systems*, vol. 54, pp. 128–136, 2013.

[15] B. Qian, Q.-Q. Wang, R. Hu, Z.-J. Zhou, C.-Q. Yu, and Z.-G. Zhou, "An effective soft computing technology based on belief-rule-base and particle swarm optimization for tipping paper permeability measurement," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 3, pp. 841–850, 2019.

[16] A. Zamuda and J. Brest, "Self-adaptive control parameters randomization frequency and propagations in differential evolution," *Swarm and Evolutionary Computation*, vol. 25, pp. 72–99, 2015.

[17] R. Tanabe and A. Fukunaga, "Success-history based parameter adaptation for differential evolution," in *2013 IEEE Congress on Evolutionary computation*, 2013, pp. 71–78.

[18] A. W. Mohamed and A. K. Mohamed, "Adaptive guided differential evolution algorithm with novel mutation for numerical optimization," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 2, pp. 253–277, 2019.

[19] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *Soft Computing*, vol. 9, no. 6, pp. 448–462, 2005.

[20] M. Leon and N. Xiong, "Greedy adaptation of control parameters in differential evolution for global optimization problems," in *2015 IEEE Congress on Evolutionary Computation (CEC)*, 2015, pp. 385–392.

[21] R. Tanabe and A. S. Fukunaga, "Improving the search performance of shade using linear population size reduction," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2014, pp. 1658–1665.

[22] E. Zorarpacı and S. A. Özel, "A hybrid approach of differential evolution and artificial bee colony for feature selection," *Expert Systems with Applications*, vol. 62, pp. 91–103, 2016.

[23] P. Rakshit, A. Konar, S. Das, L. C. Jain, and A. K. Nagar, "Uncertainty management in differential evolution induced multiobjective optimization in presence of measurement noise," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 7, pp. 922–937, 2013.

[24] M. S. Hossain, S. Rahaman, A.-L. Kor, K. Andersson, and C. Pattinson, "A belief rule based expert system for datacenter PUE prediction under uncertainty," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 2, pp. 140–153, 2017.

[25] D. Vucetic and S. P. Simonovic, "Evaluation and application of fuzzy differential evolution approach for benchmark optimization and reservoir operation problems," *Journal of Hydroinformatics*, vol. 15, no. 4, pp. 1456–1473, 2013.

[26] M. Kuhn and K. Johnson, *Applied predictive modeling*. Springer, 2013.