

Weights-Learning for Weighted Fuzzy Rule Interpolation in Sparse Fuzzy Rule-Based Systems

Shyi-Ming Chen and Yu-Chuan Chang

Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology,
Taipei, Taiwan, R. O. C.

e-mail: smchen@mail.ntust.edu.tw, D9315003@mail.ntust.edu.tw

Abstract—In this paper, we present a weights-learning algorithm based on the CHC algorithm, which is a specialization of traditional genetic algorithms, to automatically learn the optimal weights of the antecedent variables of the fuzzy rules for the proposed weighted fuzzy interpolative reasoning method based on bell-shaped membership functions. We also apply the proposed method to deal with the truck backer-upper control problem. The experimental results show that the proposed method using the optimally learned weights gets better accuracy rates than the existing methods for dealing with the truck backer-upper control problem.

Keywords—Fuzzy interpolative reasoning; sparse fuzzy rule-based systems; weighted antecedent variables; genetic algorithms

I. INTRODUCTION

Sparse fuzzy rule bases contain blank areas uncovered by fuzzy rules in the universe of discourse with respect to the input variables. When the observation occurs in these blank areas, traditional fuzzy rule-based systems may get improper fuzzy reasoning results. Because fuzzy interpolation techniques can make inferences with sparse fuzzy rule bases effectively, fuzzy interpolation is an essential solution for inference in sparse fuzzy rule-based systems. Some fuzzy interpolative reasoning methods have been presented for sparse fuzzy rule-based systems [1]–[7], [10], [11], [13], [14], [17], [18].

In this paper, we present a weights-learning algorithm based on the CHC algorithm [8], which is a specialization of traditional genetic algorithms (GAs) [12], to automatically learn the optimal weights of the antecedent variables of the fuzzy rules for the proposed weighted fuzzy interpolative reasoning method, based on bell-shaped membership functions. We also apply the proposed method to deal with the truck backer-upper control problem. The experimental results show that the proposed method using the optimally learned weights gets smaller error rates than the existing methods [1], [6], [7], [11], [15].

II. PRELIMINARIES

A bell-shaped fuzzy set A in the universe of discourse U

can be represented by $e^{-\left(\frac{x-c}{a}\right)^{2b}}$, where the parameters a and b alter the width of the membership function curve of A , the parameter c denotes the mean (i.e., the center) of the membership function of A and $b \geq 0$. For simplicity, a bell-shaped fuzzy set A can be represented by (a, b, c) . For example, Figure 2 shows a bell-shaped fuzzy set A , where $A = (2, 2, 5)$.

The representative value “ $Rep(A)$ ” of a bell-shaped fuzzy set A is defined as follows:

$$Rep(A) = c. \quad (1)$$

Let c_A and c_B be the representative values of two bell-shaped fuzzy sets A and B , respectively. If $c_A \leq c_B$, then A is at the left-hand side of B and the total ordering relationship between A and B is denoted by “ $A \prec B$ ”. Otherwise, A is at the right-hand side of B and the total ordering relationship between A and B is denoted by “ $B \prec A$ ”.

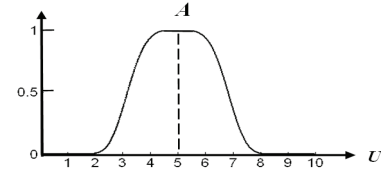


Figure 1. Bell-shaped fuzzy set $A = (2, 2, 5)$.

Generally speaking, the larger the values of the parameters a_A and b_A of the bell-shaped fuzzy set A , the larger the fuzziness of the fuzzy set A . Let A and B be two bell-shaped fuzzy sets, where $A = (a_A, b_A, c_A)$ and $B = (a_B, b_B, c_B)$. The ratios of the fuzziness $RF_a(A, B)$ and $RF_b(A, B)$ of the bell-shaped fuzzy set A to the fuzziness of the bell-shaped fuzzy set B with respect to the parameters a and b , respectively, are defined as follow:

$$RF_a(A, B) = \frac{a_A}{a_B}, \quad (2)$$

$$RF_b(A, B) = \frac{b_A}{b_B}. \quad (3)$$

III. A WEIGHTED FUZZY INTERPOLATIVE REASONING METHOD FOR SPARSE FUZZY RULE-BASED SYSTEMS USING BELL-SHAPED MEMBERSHIP FUNCTIONS

In this section, we present a weighted fuzzy interpolative reasoning method for sparse fuzzy rule-based systems using bell-shaped membership functions. Consider the following weighted multiple fuzzy rules interpolation scheme using bell-shaped membership functions:

Rule 1: If X_1 is $A_{11}(w_1)$ and X_2 is $A_{12}(w_2)$ and ...and X_m is $A_{1m}(w_m)$ Then Y is B_1

Rule 2: If X_1 is $A_{21}(w_1)$ and X_2 is $A_{22}(w_2)$ and ...and X_m is $A_{2m}(w_m)$ Then Y is B_2

⋮

Rule n: If X_1 is $A_{n1}(w_1)$ and X_2 is $A_{n2}(w_2)$ and ...and X_m is $A_{nm}(w_m)$ Then Y is B_n

Observation: X_1 is A_1^* and X_2 is A_2^* and ...and X_m is A_m^*

Conclusion: Y is B^*

where X_j denotes the j th antecedent variable, Y denotes the consequence variable, the antecedent fuzzy set A_{ij} of *Rule i* is represented by $(a_{A_{ij}}, b_{A_{ij}}, c_{A_{ij}})$, the consequence fuzzy set B_i of *Rule i* is represented by $(a_{B_i}, b_{B_i}, c_{B_i})$, the j th observation fuzzy set A_j^* is represented by $(a_{A_j^*}, b_{A_j^*}, c_{A_j^*})$, the interpolated consequence fuzzy set B^* is represented by $(a_{B^*}, b_{B^*}, c_{B^*})$, w_j denotes the weight of the antecedent variable X_j , $0 \leq w_j \leq 1$, $1 \leq j \leq m$, m denotes the number of antecedent variables appearing in the fuzzy rules and n denotes the number of fuzzy rules. Based on the weighted multiple fuzzy rules interpolation scheme, the proposed method for fuzzy rule interpolation using bell-shaped membership functions is now presented as follows:

Step 1: Choose p neighboring fuzzy rules *Rule 1*, *Rule 2*, ..., and *Rule p* with respect to the observation as the closest fuzzy rules [5] with respect to the observation, where each antecedent fuzzy set A_{ij} of the closest fuzzy rule *Rule i* is either the left nearest fuzzy set or the right nearest fuzzy set of the observation fuzzy set A_j^* , $1 \leq j \leq m$ and $1 \leq p \leq n$.

Step 2: Based on [5], calculate the weight W_i of *Rule i*, where $1 \leq i \leq p$. Assume that $A_{l_{ij}}$ and $A_{r_{ij}}$ are the left nearest antecedent fuzzy set and the right nearest antecedent fuzzy set with respect to the observation fuzzy set A_j^* , respectively, where $1 \leq l_j \leq p$ and $1 \leq r_j \leq p$. Let

$$\lambda_{l_{ij}} = \begin{cases} \left(1 - \frac{|c_{A_j^*} - c_{A_{l_{ij}}}|}{|c_{A_{r_{ij}}} - c_{A_{l_{ij}}}|}\right), & \text{if } A_{l_{ij}} < A_j^* < A_{r_{ij}}, \\ 1, & \text{if } A_j^* < A_{l_{ij}}, \text{ where } 1 \leq i \leq p, \end{cases} \quad (4)$$

$$\lambda_{r_{ij}} = \begin{cases} \left(1 - \frac{|c_{A_{r_{ij}}} - c_{A_j^*}|}{|c_{A_{r_{ij}}} - c_{A_{l_{ij}}}|}\right), & \text{if } A_{l_{ij}} < A_j^* < A_{r_{ij}}, \\ 1, & \text{if } A_j^* < A_{r_{ij}}, \text{ where } 1 \leq i \leq p, \end{cases} \quad (5)$$

where $1 \leq j \leq m$. Let

$$W_i = \frac{\min_{1 \leq j \leq m} w_j \lambda_{ij}}{\sum_{q=1}^p \min_{1 \leq j \leq m} w_j \lambda_{iq}}, \quad (6)$$

where W_i denotes the weight of the fuzzy rule *Rule i*, $0 \leq W_i \leq 1$, $\sum_{i=1}^p W_i = 1$ and $1 \leq i \leq p$.

Step 3: Calculate the mean c_{B^*} of the interpolated consequence fuzzy set B^* , shown as follows:

$$c_{B^*} = \sum_{i=1}^p W_i \times c_{B_i}, \quad (7)$$

where $0 \leq W_i \leq 1$, $\sum_{i=1}^p W_i = 1$ and $1 \leq i \leq p$.

Step 4: Calculate the ratio RF_b of the composite fuzziness of the observation fuzzy sets to the composite fuzziness of the antecedent fuzzy sets with respect to the parameter b , shown as follows:

$$CF_b^* = \sum_{j=1}^m w_j \times b_{A_j^*}, \quad (8)$$

$$CF_{b,i} = \sum_{j=1}^m w_j \times b_{A_{ij}}, \quad \text{where } 1 \leq i \leq p \quad (9)$$

$$RF_b = \frac{CF_b^*}{\sum_{i=1}^p W_i \times CF_{b,i}}, \quad (10)$$

where CF_b^* denotes the composite fuzziness of the observation fuzzy sets $A_1^*, A_2^*, \dots$, and A_m^* with respect to the parameter b , $CF_{b,i}$ denotes the composite fuzziness of the antecedent fuzzy sets $A_{i1}, A_{i2}, \dots$, and A_{im} of the fuzzy rule *Rule i* with respect to the parameter b , RF_b denotes the ratio of the composite fuzziness of the observation fuzzy sets to the composite fuzziness of the antecedent fuzzy sets A_{ij} with respect to the parameter b , $RF_b \geq 0$, $0 \leq W_i \leq 1$, $\sum_{i=1}^p W_i = 1$, $1 \leq j \leq m$ and $1 \leq i \leq p$. Calculate the value of b_{B^*} of the interpolated consequence fuzzy set $B^* = (a_{B^*}, b_{B^*}, c_{B^*})$, shown as follows:

$$b_{B^*} = RF_b \times \left(\sum_{i=1}^p W_i \times b_{B_i} \right), \quad (11)$$

where $0 \leq W_i \leq 1$, $\sum_{i=1}^p W_i = 1$ and $1 \leq i \leq p$.

Step 5: Calculate the ratio RF_a of the composite fuzziness of the observation fuzzy sets to the composite fuzziness of the antecedent fuzzy sets with respect to the parameter a , shown as follows:

$$CF_a^* = \sum_{j=1}^m w_j \times a_{A_j^*}, \quad (12)$$

$$CF_{a,i} = \sum_{j=1}^m w_j \times a_{A_{ij}}, \quad (13)$$

$$RF_a = \frac{CF_a^*}{\sum_{i=1}^p W_i \times CF_{a,i}}, \quad (14)$$

where CF_a^* denotes the composite fuzziness of the observation fuzzy sets $A_1^*, A_2^*, \dots$ and A_m^* with respect to the parameter a , $CF_{a,i}$ denotes the composite fuzziness of the antecedent fuzzy sets $A_{i1}, A_{i2}, \dots$, and A_{im} of the fuzzy rule *Rule i* with respect to the parameter a , RF_a denotes the ratio of the composite fuzziness of the observation fuzzy sets A_j^* to the composite fuzziness of the antecedent fuzzy sets A_{ij} with respect to the parameter a , $RF_a \geq 0$, $0 \leq W_i \leq 1$, $\sum_{i=1}^p W_i = 1$, $1 \leq j \leq m$ and $1 \leq i \leq p$. Calculate the value of a_{B^*} of the interpolated consequence fuzzy set $B^* = (a_{B^*}, b_{B^*}, c_{B^*})$, where

$$a_{B^*} = RF_a \times \left(\sum_{i=1}^p W_i \times a_{B_i} \right), \quad (15)$$

$0 \leq W_i \leq 1$, $\sum_{i=1}^p W_i = 1$ and $1 \leq i \leq p$. Therefore, we can get the

interpolated consequence fuzzy set B^* , where $B^* = (a_{B^*}, b_{B^*}, c_{B^*})$.

IV. A WEIGHTS-LEARNING ALGORITHM TO AUTOMATICALLY LEARN THE OPTIMAL WEIGHTS OF THE ANTECEDENT VARIABLES OF THE FUZZY RULES

In this section, we present a weights-learning algorithm based on the CHC algorithm [8], which is a specialization of traditional genetic algorithms (GAs) [12], to automatically learn the optimal weights of the antecedent variables of the fuzzy rules for the weighted fuzzy interpolative reasoning method. The CHC algorithm [8] uses the crossover generation, elitist selection, heterogeneous recombination and cataclysmic mutation for evolution. Assume that a fuzzy rule-based system has a knowledge base consisting of n fuzzy rules, where $X_1, X_2, \dots, X_m$ denote the antecedent variables of the fuzzy rules with respect to the inputs $x_1, x_2, \dots, x_m$, respectively, and Y denotes the consequence variable of the fuzzy rules with respect to the output. Let W_{X_j} denote the weight of the antecedent variable X_j , where $1 \leq j \leq m$. Given a set S of training samples, where $S = \{S_1, S_2, \dots, S_P\}$, the input values of each training sample S_i are represented by $(x_{i1}, x_{i2}, \dots, x_{im})$, the target output of training sample S_i is represented by y_i , $1 \leq i \leq P$ and P denotes the number of training samples. In the proposed weights-learning algorithm, a population consists of l chromosomes and each chromosome consists of m genes, where the value g_j of the j th gene denotes the weight W_{X_j} of the antecedent variable X_j of the fuzzy rules, $0 \leq g_j \leq 1$ and $1 \leq j \leq m$, as shown in Figure 2. Assume that h denotes the heterogeneous recombination threshold for incest prevention.

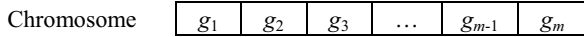


Figure 2. Encoding the weights of the antecedent variables into a chromosome.

The proposed weights-learning algorithm is now presented as follows:

Step 1: Initialize two chromosomes, where one chromosome has the weights 0, 1, 0, 1, ..., 0, 1 and the other chromosome has the weights 1, 0, 1, 0, ..., 1, 0. Then, the system randomly generates the remaining $l - 2$ chromosomes to form the initial population. Let $h = \left\lceil \frac{m}{4} \right\rceil$.

Step 2: For each chromosome in the current population, decode the chromosome into the weight W_{X_j} of each antecedent variable X_j of the fuzzy rules, where $1 \leq j \leq m$. Based on the weighted fuzzy rules interpolation scheme, let the input values $x_{i1}, x_{i2}, \dots, x_{im}$ of each training sample $S_i (x_{i1}, x_{i2}, \dots, x_{im}, y_i)$ be the input observation " $X_1 = x_{i1}, X_2 = x_{i2}, \dots, X_m = x_{im}$ ", where $1 \leq i \leq P$ and y_i denotes the target output of S_i . For each

training sample $S_i = (x_{i1}, x_{i2}, \dots, x_{im}, y_i)$, apply the proposed weighted fuzzy interpolative reasoning method to generate the fuzzy interpolated result B_i^* with respect to the input observation " $X_1 = x_{i1}, X_2 = x_{i2}, \dots, X_m = x_{im}$ ", where $1 \leq i \leq P$. For each B_i^* , if B_i^* is a fuzzy set, then use the center of gravity defuzzification method [16] to defuzzify B_i^* into a crisp-valued inferred output y_i^* , where $1 \leq i \leq P$. Based on a predefined fitness function, calculate the fitness value of each chromosome between the inferred outputs and the target outputs of the training samples.

Step 3: Select chromosomes from the current population based on the binary tournament selection method [9] to form a mating pool. The system repeatedly randomly selects two chromosomes from the current population and puts the chromosome having the larger fitness value into the mating pool and then puts the two selected chromosomes back into the current population. Repeatedly perform the above process until the number of chromosomes in the mating pool is equal to l , where l denotes the number of chromosomes of a population.

Step 4: Perform the blend crossover (BLX- α) method [8] with the chromosomes in the mating pool, where $0 \leq \alpha < 1$ (Note: In this paper, we let $\alpha = 0.5$). For each chromosome C_i in the mating pool, the system randomly generates a real value between 0 and 1. If the randomly generated real value is smaller than a predefined crossover rate γ where $\gamma \in [0, 1]$, then the system randomly chooses another chromosome C_j from the mating pool in which the number of distinct genes between C_i and C_j is larger than the value of h , and randomly chooses $\left\lceil \frac{m}{4} \right\rceil$ crossover points to perform the BLX- α crossover

[8] with the selected chromosomes C_i and C_j . Figure 3 shows an example how the system updates the genes between the two selected chromosomes C_i and C_j , where the values of the genes g_2 and g_7 of C_i and C_j shown in Figure 3 are updated by the system, respectively.

		Crossover Point					Crossover Point			
		↓					↓			
		g_1	g_2	g_3	g_4	g_7	g_8	g_9	g_{10}	
Before	C_i	0.58	0.12	0.80	0.42	0.13	0.21	0.48	0.62	
Performing										
Crossover	C_j	0.58	0.30	0.85	0.19	0.07	0.51	0.24	0.17	
Operation										
		g_1	g_2	g_3	g_4	g_7	g_8	g_9	g_{10}	
After	C_i	0.58	0.25	0.80	0.42	0.13	0.21	0.40	0.62	
Performing										
Crossover	C_j	0.58	0.165	0.85	0.19	0.07	0.51	0.36	0.17	
Operation										

Figure 3. The system updates the genes between the two selected chromosomes.

Step 5: For each chromosome in the mating pool, decode the chromosome into the weight W_{X_j} of each antecedent variable X_j of the fuzzy rules, where $1 \leq j \leq m$. Based on the weighted fuzzy rules interpolation scheme, let the input values $x_{i1}, x_{i2}, \dots, x_{im}$ of each training sample $S_i = (x_{i1}, x_{i2}, \dots, x_{im}, y_i)$ be the input observation " $X_1 = x_{i1}, X_2 = x_{i2}, \dots, X_m = x_{im}$ ", where $1 \leq i \leq P$ and y_i denotes the target output of S_i . For each training sample $S_i = (x_{i1}, x_{i2}, \dots, x_{im}, y_i)$, apply the proposed weighted

fuzzy interpolative reasoning method to generate the fuzzy interpolated result B_i^* with respect to the input observation " $X_1 = x_{i1}, X_2 = x_{i2}, \dots, X_m = x_{im}$ ", where $1 \leq i \leq P$. For each B_i^* , if B_i^* is a fuzzy set, then use the center of gravity defuzzification method [16] to defuzzify B_i^* into a crisp-valued inferred output y_i^* , where $1 \leq i \leq P$. Based on a predefined fitness function, calculate the fitness value of each chromosome between the inferred outputs and the target outputs of the training samples. Perform the elitist replacement to form the population of the next generation. The system selects chromosomes having the top l largest fitness values from the l chromosomes in the current population together with the l chromosomes in the mating pool to form the population of the next generation. If the number of selected chromosomes from the mating pool is equal to zero, then let $h = h - 1$. Otherwise, go to Step 6. If $h = 0$, then go to Step 7. Otherwise, go to Step 6.

Step 6: If the number of evolutions meets a predefined maximum number of evolutions, then let the chromosome, which has the largest fitness value be the optimal chromosome and go to Step 8. Otherwise, go to Step 2.

Step 7: Reinitialize the population using the cataclysmic mutation method [8]. The system puts the best chromosome having the largest fitness value in the current population into the population of the next generation and mutates the remaining $l - 1$ chromosomes of the current population to form the $l - 1$ chromosomes in the population of the next generation. For each gene of a mutated chromosome, the system randomly generates a real value between 0 and 1. If the generated real value is smaller than 0.5, then the system replaces the value of the gene with a randomly generated real value between 0 and 1. Let $h = \left\lceil \frac{m}{4} \right\rceil$, where m denotes the number of genes in a chromosome, and go to Step 2.

Step 8: Decode the chromosome having the largest fitness value into the weights $W_{X_1}, W_{X_2}, \dots$, and W_{X_m} to be used in the proposed weighted fuzzy interpolative reasoning method.

V. EXPERIMENTAL RESULTS

In this section, we apply the proposed weights-learning algorithm to automatically learn the optimal weights of the antecedent variables of the fuzzy rules for the weighted fuzzy interpolative reasoning method to deal with the truck backer-upper control problem [15].

The goal of the truck backer-upper control problem [15] is to back up a truck to a loading dock as precisely as possible, where the behavior of the truck can be described by three state variables $x \in [0, 100]$, $y \in [0, 200]$ and $\phi \in [-90, 270]$, x denotes the horizontal axis, y denotes the vertical axis and ϕ denotes the azimuth angle. The truck starts from an initial state and should back up to the loading dock with the target state (50, 200, 90), i.e., $x = 50$, $y = 200$, and $\phi = 90$. In order to control the truck, the steering angle $\theta \in [-30, 30]$ is provided for each move of the truck. In general, it is assumed that the distance between the truck and the loading dock is quite large, and the state variable y can be ignored, so that the truck backer-upper control problem can be simplified into $\theta = f(x, \phi)$ [11]. In [11],

Huang and Shen used a more compact fuzzy rule base of the six fuzzy rules shown in Figure 4 for truck backer-upper control. In this paper, we adopt the fuzzy rule base of the six fuzzy rules shown in Figure 4 for truck backer-upper control, where each row shown in Figure 4 denotes a fuzzy rule, shown as follows:

If x is A and ϕ is B Then θ is C ,

where x and ϕ are the antecedent variables denoting the horizontal axis and the azimuth angle, respectively, θ is the consequence variable denoting the steering angle, and A, B and C are the possible fuzzy sets of x, ϕ and θ , respectively.

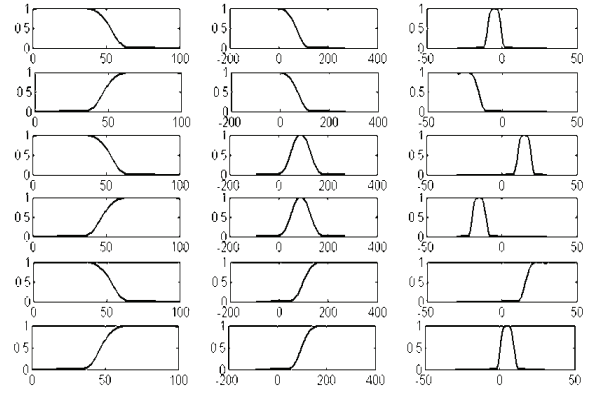


Figure 4. Membership functions of the six fuzzy rules [11].

Based on the weighted multiple fuzzy rules interpolation scheme, assume that each fuzzy rule of the six fuzzy rules for truck backer-upper control is represented as follows:

If x is $A(w_x)$ and ϕ is $B(w_\phi)$ Then θ is C ,

where x and ϕ are the antecedent variables denoting the horizontal axis and the azimuth angle, respectively, θ is the consequence variable denoting the steering angle, A, B and C are the possible fuzzy sets of x, ϕ and θ , respectively, and w_x and w_ϕ denote the weights of the antecedent variables x and ϕ , respectively. Let each chromosome have two genes g_1 and g_2 which denote the weights w_x and w_ϕ of the antecedent variables x and ϕ , respectively, where $0 \leq g_1 \leq 1$ and $0 \leq g_2 \leq 1$. Based on the fuzzy rule base shown in Figure 4 and the 64 initial states shown in Table I, we apply the proposed weights-learning algorithm for truck backer-upper control shown in Figure 5 to learn the optimal weights of w_x and w_ϕ for the proposed weighted fuzzy interpolative reasoning method, where the population size is 30, the crossover rate is 0.95 and the number of evolutions is 100. In this paper, we use the following fitness function to calculate the fitness value $Fitness$ of each chromosome for truck backer-upper control:

$$Fitness = \left(\frac{1}{64} \sum_{i=1}^{64} \frac{|x_i^f - 50| + |\phi_i^f - 90|}{2} \right)^{-1}, \quad (16)$$

where $1 \leq i \leq 64$.

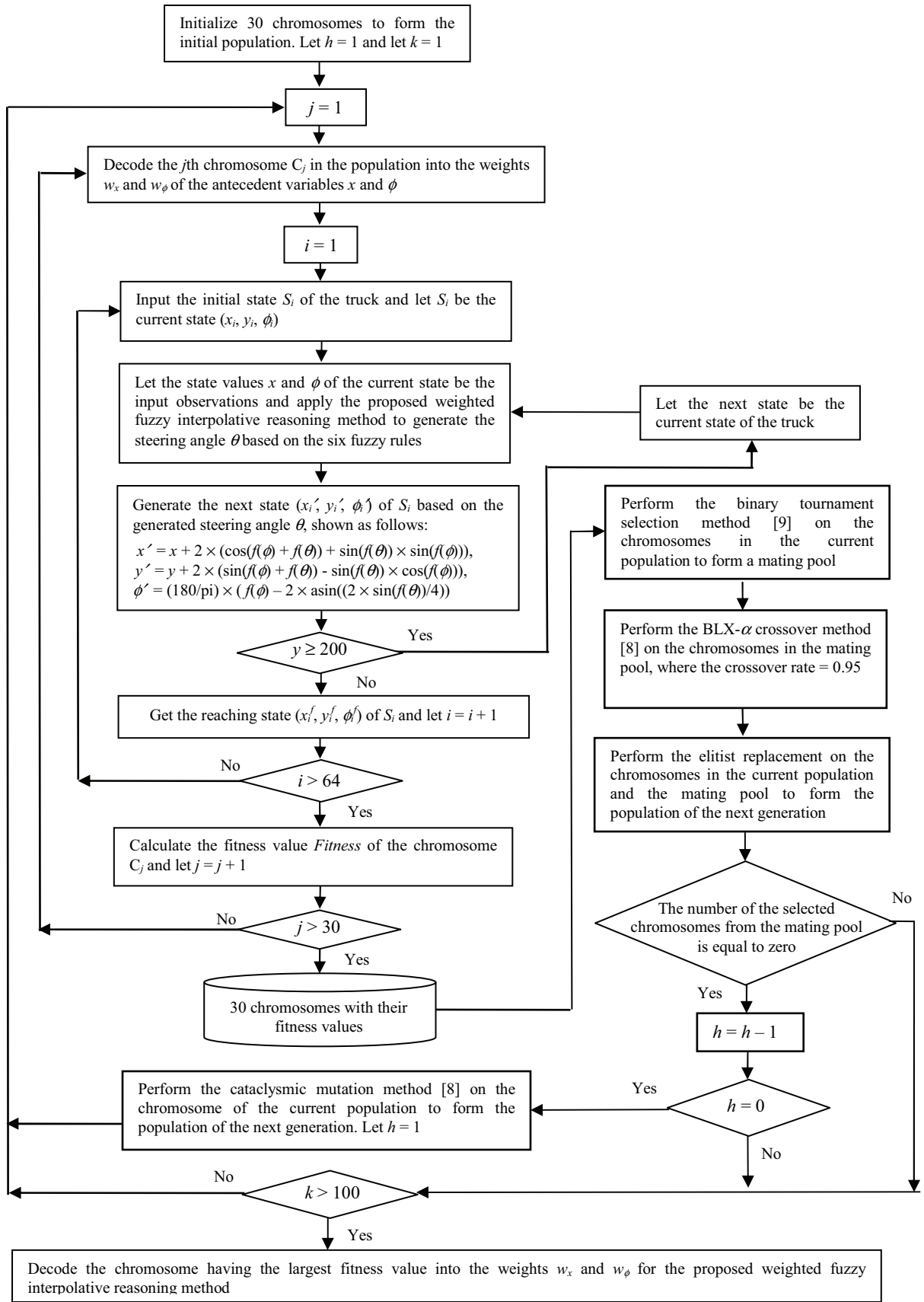


Figure 5. The flowchart of the proposed weights-learning algorithm for truck backer-upper control.

TABLE I
64 INITIAL STATES WITH RESPECT TO THE TARGET STATE (50, 200, 90)

Initial States S_i	(x_i^s, y_i^s, ϕ_i^s)	Initial States S_i	(x_i^s, y_i^s, ϕ_i^s)	Initial States S_i	(x_i^s, y_i^s, ϕ_i^s)
S_1	(10, 10, 270)	S_{23}	(30, 70, 0)	S_{45}	(60, 50, 90)
S_2	(10, 20, 225)	S_{24}	(30, 80, -45)	S_{46}	(60, 60, 45)
S_3	(10, 30, 180)	S_{25}	(40, 10, 270)	S_{47}	(60, 70, 0)
S_4	(10, 40, 135)	S_{26}	(40, 20, 225)	S_{48}	(60, 80, -45)
S_5	(10, 50, 90)	S_{27}	(40, 30, 180)	S_{49}	(70, 10, 270)
S_6	(10, 60, 45)	S_{28}	(40, 40, 135)	S_{50}	(70, 20, 225)
S_7	(10, 70, 0)	S_{29}	(40, 50, 90)	S_{51}	(70, 30, 180)
S_8	(10, 80, -45)	S_{30}	(40, 60, 45)	S_{52}	(70, 40, 135)
S_9	(20, 10, 270)	S_{31}	(40, 70, 0)	S_{53}	(70, 50, 90)
S_{10}	(20, 20, 225)	S_{32}	(40, 80, -45)	S_{54}	(70, 60, 45)
S_{11}	(20, 30, 180)	S_{33}	(50, 10, 270)	S_{55}	(70, 70, 0)
S_{12}	(20, 40, 135)	S_{34}	(50, 20, 225)	S_{56}	(70, 80, -45)
S_{13}	(20, 50, 90)	S_{35}	(50, 30, 180)	S_{57}	(80, 10, 270)
S_{14}	(20, 60, 45)	S_{36}	(50, 40, 135)	S_{58}	(80, 20, 225)
S_{15}	(20, 70, 0)	S_{37}	(50, 50, 90)	S_{59}	(80, 30, 180)
S_{16}	(20, 80, -45)	S_{38}	(50, 60, 45)	S_{60}	(80, 40, 135)
S_{17}	(30, 10, 270)	S_{39}	(50, 70, 0)	S_{61}	(80, 50, 90)
S_{18}	(30, 20, 225)	S_{40}	(50, 80, -45)	S_{62}	(80, 60, 45)
S_{19}	(30, 30, 180)	S_{41}	(60, 10, 270)	S_{63}	(80, 70, 0)
S_{20}	(30, 40, 135)	S_{42}	(60, 20, 225)	S_{64}	(80, 80, -45)
S_{21}	(30, 50, 90)	S_{43}	(60, 30, 180)		
S_{22}	(30, 60, 45)	S_{44}	(60, 40, 135)		

Based on the proposed weights-learning algorithm for truck backer-upper control shown in Figure 5, we can get the optimally learned weights $w_x = 0.55$ and $w_\phi = 0.70$ with respect to the antecedent variables x and ϕ , respectively. Table II shows a comparison of the reaching states of the four trajectories for different methods with respect to the target state (50, 200, 90). From Table II, we can see that the proposed method using the optimally learned weights $w_x = 0.55$ and $w_\phi = 0.70$ gets better truck backer-upper control results than the ones using the traditional fuzzy inference system [15], the CCL method [1], the CK method [6], the CKCP method [7] and the HS method [11].

TABLE II
A COMPARISON OF REACHING STATES OF THE FOUR TRAJECTORIES FOR DIFFERENT METHODS WITH RESPECT TO THE TARGET STATE (50, 200, 90)

Initial States	(20, 20, 90)	(80, 30, 120)	(60, 40, -90)	(10, 30, 220)
9 Rules without Interpolation [15]	(53.35, 200, 89.69)	(53.45, 200, 90.52)	(53.37, 200, 90.35)	(53.57, 200, 90.58)
6 Rules without Interpolation [15]	(53.44, 200, 89.51)	(53.40, 200, 90.45)	(53.43, 200, 90.84)	(53.48, 200, 90.84)
6 Rules with the HS Method [11]	(49.68, 200, 84.65)	(49.49, 200, 84.83)	(49.84, 200, 97.97)	(49.71, 200, 97.98)
6 Rules with the CCL Method [1]	(41.42, 200, 84.61)	(56.50, 200, 89.94)	(63.44, 200, 94.34)	(44.5, 200, 86.94)
6 rules with the CK Method [6]	(50.39, 200, 88.36)	(50.57, 200, 89.34)	(50.58, 200, 89.94)	(50.54, 200, 90.00)
6 rules with the CKCP Method [7]	(49.99, 200, 89.96)	(50.01, 200, 90.05)	(50.03, 200, 90.05)	(50.00, 200, 89.99)
6 Rules with the Proposed Method Using the Optimally Learned Weights $w_x = 0.55$ and $w_\phi = 0.70$	(49.9997, 200, 89.9997)	(50.0026, 200, 90.0081)	(50.0028, 200, 90.0064)	(49.9988, 200, 89.9974)

VI. CONCLUSION

In this paper, we have presented a weights-learning algorithm based on the CHC algorithm to automatically learn the optimal weights of the antecedent variables of the fuzzy rules for the weighted fuzzy interpolative reasoning method. We also apply the proposed method to deal with the truck backer-upper control problem. The experimental results show

that the proposed method using the optimally learned weights gets better accuracy than the existing methods [1], [6], [7], [11], [15] for dealing with the truck backer-upper control problem.

ACKNOWLEDGMENT

This work was supported in part by the National Science Council, Republic of China, under Grant NSC 97-2221-E-011-107-MY3.

REFERENCES

- [1] Y. C. Chang, S. M. Chen, and C. J. Liao, "Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on the areas of fuzzy sets," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 5, pp. 1285-1301, 2008.
- [2] Y. C. Chang and S. M. Chen, "A new method for multiple fuzzy rules interpolation with weighted antecedent variables," in *Proc. 2008 IEEE Int. Conf. Systems, Man and Cybernetics*, Singapore, 2008, pp. 76-81.
- [3] Y. C. Chang and S. M. Chen, "A new fuzzy interpolative reasoning method based on interval type-2 fuzzy sets," in *Proc. 2008 IEEE Int. Conf. Systems, Man and Cybernetics*, Singapore, 2008, pp. 82-87.
- [4] S. M. Chen and Y. C. Chang, "A new weighted fuzzy rule interpolation method based on GA-based weights-learning techniques," in *Proc. 2010 International Conference on Machine Learning and Cybernetics*, Qingdao, Shandong, China, 2010, pp. 2705-2711.
- [5] S. M. Chen and Y. C. Chang, "A new method for weighted fuzzy interpolative reasoning based on weights-learning techniques," in *Proc. 2010 IEEE International Conference on Fuzzy Systems*, Barcelona, Spain, 2010, pp. 339-344.
- [6] S. M. Chen and Y. K. Ko, "Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on α -cuts and transformation techniques," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 6, pp. 1626-1648, 2008.
- [7] S. M. Chen, Y. K. Ko, Y. C. Chang, and J. S. Pan, "Weighted fuzzy interpolative reasoning based on weighted incremental transformation and weighted ratio transformation techniques," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 6, pp. 1412-1427, 2009.
- [8] L. Eshelman, "The CHC adaptive search algorithm: How to have safe search when engaging in nontraditional genetic recombination," *Foundations of Genetic Algorithms*, pp. 265-283, San Mateo, Morgan Kaufmann, 1991.
- [9] D. E. Goldberg and K. Deb, "A comparison analysis of selection schemes used in genetic algorithms," *Foundations of Genetic Algorithms*, pp. 69-93, San Mateo, Morgan Kaufmann, 1991.
- [10] Z. H. Huang and Q. Shen, "Fuzzy interpolative reasoning via scale and move transformations," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 2, pp. 340-359, 2006.
- [11] Z. H. Huang and Q. Shen, "Fuzzy interpolation and extrapolation : A practical approach," *IEEE Trans. Fuzzy Syst.*, vol. 16, no. 1, pp. 13-28, 2008.
- [12] J. H. Holland, *Adaptation in Natural and Artificial Systems*, Cambridge, MA: MIT Press, 1975.
- [13] L. T. Koczy and K. Hirota, "Approximate reasoning by linear rule interpolation and approximation," *Int. J. Approx. Reasoning*, vol. 9, no. 3, pp. 197-225, 1993.
- [14] Y. M. Li, D. M. Huang, E. C. C. Tsang, and L. N. Zhang, "Weighted fuzzy interpolative reasoning method," in *Proc. 2005 IEEE Int. Conf. Machine Learning and Cybern.*, Guangzhou, China, 2005, pp. 18-21.
- [15] A. Lotfi, "Fuzzy inference system toolbox for matlab (FISMAT)," 2000 [Online]. Available: <http://www.lotfi.net/research/fismat/index.html>.
- [16] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*. NY: McGraw Hill, 1996.
- [17] Y. Yam, M. L. Wong, and P. Baranyi, "Interpolation with function space representation of membership functions," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 3, pp. 398-411, 2006.
- [18] L. Yang and Q. Shen, "Towards adaptive interpolative reasoning," in *Proc. 2009 IEEE International Conference on Fuzzy Systems*, Korea, 2009, pp. 542-549.