

Temperature Prediction Based on Genetic Simulated Annealing Techniques and High-Order Fuzzy Time Series

Li-Wei Lee*, Li-Hui Wang**, and Shyi-Ming Chen*

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

**Department of Finance, Chihlee Institute of Technology, Panchiao, Taipei, Taiwan, R. O. C.

Abstract

In this paper, we present a new approach for temperature prediction based on genetic simulated annealing techniques and high-order fuzzy time series, where the simulated annealing techniques are used to deal with the mutation operations of genetic algorithms. The proposed method uses genetic simulated annealing techniques to adjust the length of each interval in the universe of discourse to increase the forecasting accuracy rate. It can get a higher forecasting accuracy rate than the existing methods.

1 Introduction

It is obvious that people are always interested in oncoming events, where an event may be affected by many factors. If we can consider these many factors for dealing with forecasting problems, then we can get a higher forecasting accuracy rate. In recent years, some methods have been presented for dealing with forecasting problems [1]-[15].

In this paper, we present a new method for temperature prediction based on genetic simulated annealing techniques and high-order fuzzy time series, where the simulated annealing techniques [16] are used to deal with mutation operations of genetic algorithms. The proposed method uses genetic simulated annealing techniques to adjust the length of each interval in the universe of discourse for increasing the forecasting accuracy rate. It can get a higher forecasting accuracy rate than the existing methods.

2 Fuzzy Time Series

In [10]-[12], Song and Chissom presented the concepts of fuzzy time series based on the fuzzy set theory [17]. In the following, we briefly review the definitions of fuzzy time series from [3], [11] and [13].

Definition 2.1: Let $Y(t)$ ($t = \dots, 0, 1, 2, \dots$) be the universe of discourse and be a subset of R . Assume that $f_i(t)$ ($i = 1, 2, \dots$) is defined in the universe of discourse $Y(t)$, and assume that $F(t)$ is a collection of $f_i(t)$ ($i = 1, 2, \dots$), then $F(t)$ is called a fuzzy time series of $Y(t)$ ($t = \dots, 0, 1, 2, \dots$).

Definition 2.2: Let $F(t)$ be a fuzzy time series. If $F(t)$ is caused by $F(t-1)$, $F(t-2)$, \dots , and $F(t-n)$, then the n th-order fuzzy logical relationship is represented by

$$F(t-n), \dots, F(t-2), F(t-1) \rightarrow F(t), \quad (1)$$

where $F(t-n)$, \dots , $F(t-2)$, $F(t-1)$ and $F(t)$ are fuzzy sets, “ $F(t-n)$, \dots , $F(t-2)$, $F(t-1)$ ” is called the current state of the n th-order fuzzy logical relationship, and $F(t)$ is called the next state of the n th-order fuzzy logical relationship.

3 Basic Concepts of Simulated Annealing Algorithms and Genetic Algorithms

In [16], Kirkpatrick et al. proposed the simulated annealing algorithm. A simulated annealing algorithm takes into account not only downhill move, but also permits uphill moves with an assigned probability depending on the “state temperature”. The basic concept of a simulated annealing algorithm is derived by observing the change of energy in a process in which materials solidify from the liquid state to the solid state. When the system’s temperature decreases gradually in the annealing schedule, if the energy of the material in a new state is lower than the energy of the material in the current state, then the system will replace the current state by the new state. Otherwise, whether the new state can be accepted or not depends on the probability p shown as follows:

$$p = e^{\frac{-\Delta E}{kT}}, \quad (2)$$

where p denotes the probability that the system accepts the new state, T denotes the current system temperature, k denotes the Boltzmann’s constant, and ΔE denotes the difference between the energy of the new state and the energy of the current state.

The concept of genetic algorithms was proposed by Holland [18], where a population consists of chromosomes and a chromosome consists of genes. The number of chromosomes in a population is called the “population size”. The reproduction operation, crossover operation and mutation operation of genetic algorithms can refer to [18], [19] and [20].

4 A New Method for Temperature Prediction Based on Genetic Simulated Annealing Techniques and High-Order Fuzzy Time Series

In this section, we present a new method for temperature prediction based on genetic simulated annealing techniques and high-order fuzzy time series.

The method is essentially a modification of the method presented in [7]. Table 1 shows the historical data of the daily average temperature from June 1996 to September 1996 in Taipei, Taiwan [21]. Table 2 shows the historical data of the daily cloud density from June 1996 to September 1996 in Taipei, Taiwan [21]. In [2], the daily average temperature is called the “Main-Factor” of the fuzzy time series and the daily cloud density is called the “Second-Factor” of the fuzzy time series. First, based on Table 1, Table 2 and [7], we define the universe of discourse of the daily average temperature $U = [23, 32]$ and define the universe of discourse of the daily cloud density $V = [0, 100]$.

Table 1. Historical Data of the Daily Average Temperature from June 1996 to September 1996 in Taipei, Taiwan (Unit: °C) [21]

Day	Month	June	July	August	September
1		26.1	29.9	27.1	27.5
2		27.6	28.4	28.9	26.8
3		29.0	29.2	28.9	26.4
4		30.5	29.4	29.3	27.5
5		30.0	29.9	28.8	26.6
6		29.5	29.6	28.7	28.2
7		29.7	30.1	29.0	29.2
8		29.4	29.3	28.2	29.0
9		28.8	28.1	27.0	30.3
10		29.4	28.9	28.3	29.9
11		29.3	28.4	28.9	29.9
12		28.5	29.6	28.1	30.5
13		28.7	27.8	29.9	30.2
14		27.5	29.1	27.6	30.3
15		29.5	27.7	26.8	29.5
16		28.8	28.1	27.6	28.3
17		29.0	28.7	27.9	28.6
18		30.3	29.9	29.0	28.1
19		30.2	30.8	29.2	28.4
20		30.9	31.6	29.8	28.3
21		30.8	31.4	29.6	26.4
22		28.7	31.3	29.3	25.7
23		27.8	31.3	28.0	25.0
24		27.4	31.3	28.3	27.0
25		27.7	28.9	28.6	25.8
26		27.1	28.0	28.7	26.4
27		28.4	28.6	29.0	25.6
28		27.8	28.0	27.7	24.2
29		29.0	29.3	26.2	23.3
30		30.2	27.9	26.0	23.5
31			26.9	27.7	

Table 2. Historical Data of the Daily Cloud Density from June 1996 to September 1996 in Taipei, Taiwan (Unit: %) [21]

Day	Month	June	July	August	September
1		36	15	100	29
2		23	31	78	53
3		23	26	68	66
4		10	34	44	50
5		13	24	56	53
6		30	28	89	63
7		45	50	71	36
8		35	34	28	76
9		26	15	70	55
10		21	8	44	31
11		43	36	48	31
12		40	13	76	25
13		30	26	50	14
14		29	44	84	45
15		30	25	69	38
16		46	24	78	24
17		55	26	39	19
18		19	25	20	39
19		15	21	24	14
20		56	35	25	3
21		60	29	19	38
22		96	48	46	70
23		63	53	41	71
24		28	44	34	70
25		14	100	29	40
26		25	100	31	30
27		29	91	41	34
28		55	84	14	59
29		29	38	28	83
30		19	46	33	38
31			95	26	

The proposed method is now presented as follows:
Step 1: Partition the universe of discourse $U = [U_{\min}, U_{\max}]$ into n intervals u_1, u_2, \dots, u_n , where $u_1 = [U_{\min}, x_1]$, $u_2 = [x_1, x_2]$, \dots , $u_n = [x_{n-1}, U_{\max}]$, U_{\min} denotes the minimum value in the universe of discourse U , U_{\max} denotes the maximum value in the universe of discourse U , and $x_1 \leq x_2 \leq \dots \leq x_{n-1}$. Partition the universe of discourse $V = [V_{\min}, V_{\max}]$ into m intervals v_1, v_2, \dots, v_m , where $v_1 = [y_1, V_{\max}]$, $v_2 = [y_2, y_1]$, \dots , $v_m = [V_{\min},$

$y_{m-1}]$, V_{\min} denotes the minimum value in the universe of discourse V , V_{\max} denotes the maximum value in the universe of discourse V , and $y_1 \geq y_2 \geq \dots \geq y_{m-1}$. Define each chromosome consisting of $n-1$ “X genes” and $m-1$ “Y genes”, where the contents of each chromosome are represented by an array $\langle x_1, x_2, \dots, x_{n-1}, y_1, y_2, \dots, y_{m-1} \rangle$, $x_1 \leq x_2 \leq \dots \leq x_{n-1}$, and $y_1 \geq y_2 \geq \dots \geq y_{m-1}$. In this paper, we assume that a population consists of 30 chromosomes and assume that the system randomly generates 30 chromosomes as the initial population.

Step 2: Define linguistic terms of the main-factor represented by fuzzy sets A_1, A_2, \dots, A_n , shown as follows:

$$A_1 = 1/u_1 + 0.5/u_2 + 0/u_3 + 0/u_4 + 0/u_5 + \dots + 0/u_{n-2} + 0/u_{n-1} + 0/u_n,$$

$$A_2 = 0.5/u_1 + 1/u_2 + 0.5/u_3 + 0/u_4 + 0/u_5 + \dots + 0/u_{n-2} + 0/u_{n-1} + 0/u_n,$$

$$A_3 = 0/u_1 + 0.5/u_2 + 1/u_3 + 0.5/u_4 + 0/u_5 + \dots + 0/u_{n-2} + 0/u_{n-1} + 0/u_n,$$

$$\vdots$$

$$A_n = 0/u_1 + 0/u_2 + 0/u_3 + 0/u_4 + 0/u_5 + \dots + 0/u_{n-2} + 0.5/u_{n-1} + 1/u_n.$$

Define linguistic terms of the second-factor represented by fuzzy sets B_1, B_2, \dots, B_m , shown as follows:

$$B_1 = 1/v_1 + 0.5/v_2 + 0/v_3 + \dots + 0/v_{m-2} + 0/v_{m-1} + 0/v_m,$$

$$B_2 = 0.5/v_1 + 1/v_2 + 0.5/v_3 + \dots + 0/v_{m-2} + 0/v_{m-1} + 0/v_m,$$

$$B_3 = 0/v_1 + 0.5/v_2 + 1/v_3 + \dots + 0/v_{m-2} + 0/v_{m-1} + 0/v_m,$$

$$\vdots$$

$$B_m = 0/v_1 + 0/v_2 + 0/v_3 + \dots + 0/v_{m-2} + 0.5/v_{m-1} + 1/v_m.$$

Based on [22], fuzzify the historical data of the main-factor and the second-factor, respectively, based on each chromosome of the population. For example, if the value of the main-factor of day i belongs to interval u_j , and fuzzy set A_j whose maximum membership value occurs at interval u_j , then the value of the main-factor of day i is fuzzified into A_j , where $1 \leq j \leq n$; if the value of the second-factor of day i belongs to interval v_s , and fuzzy set B_s whose maximum membership value occurs at v_s , then the value of the second-factor of day i is fuzzified into B_s , where $1 \leq s \leq m$.

Step 3: Construct two-factors k th-order fuzzy time series relationship groups, where $k \geq 2$.

Step 4: Forecast the values based on the principles presented in [7].

Step 5: Perform the reproduction operations based on the roulette wheel selection method [19]. In this paper, the average forecasting error rate (AFER) is used as the fitness value of each chromosome in the genetic algorithm for temperature prediction, where.

$$AFER = \frac{\sum_{i=1}^n |(\text{Forecasted Value of Day } i) - \text{Actual Value of Day } i|}{n} \times 100\%. \quad (3)$$

The smaller the fitness value (Note: The fitness value is the average forecasting error rate) of a chromosome, the higher the chance of the chromosome to be chosen for put into the mating pool. In this paper, the system chooses chromosomes from the current population into the mating pool according to their reciprocal fitness values. For example, let f_i denote the fitness value of the i th chromosome and let r_i be the reciprocal f_i , i.e., $r_i = 1/f_i$. The selected probability p_i of the i th chromosome is denoted by $p_i = r_i / \sum_j r_j$. The larger the selected

probability of a chromosome, the higher the chance of the chromosome to be chosen for put into the mating pool. Repeatedly perform the reproduction operations, until the number of chromosomes in the mating pool is the same as the number of chromosomes in the current population. Then, let the mating pool become the current population.

Step 6: Randomly select two chromosomes from the population to perform the crossover operations, until all chromosomes in the population have been selected. If the system randomly generates a real value between zero and one that is smaller than or equal to the crossover rate, then the system randomly selects a crossover point of a X gene and a crossover point of a Y gene from the two selected chromosomes of the current population to exchange genes after the crossover point. Otherwise, the selected chromosomes will not perform the crossover operation. In this paper, the crossover rate is set to 0.8. When performing the crossover operation, the system randomly selects one crossover point of X genes and one crossover point of Y genes, where the crossover point of X genes is an integer between 1 and n-1, n is the number of X genes, the crossover point of Y genes is an integer between 1 and m-1, and m is the number of Y genes. Furthermore, if the derived values of the chromosomes are not sorted by the values of genes in an ascending sequence, the system will sort the values of genes in the chromosomes in an ascending sequence.

Step 7: Use the simulated annealing mutation (SAM) algorithm shown in Fig. 1 to perform the mutation operations. For each chromosome in the population, the system generates a real value between zero and one to determine whether the system performs the simulated annealing mutation or not. If the real value generated by the system for a chromosome is smaller than or equal to the mutation rate (Note: In this paper, the mutation rate is 0.05), then the system applies the simulated annealing mutation algorithm to perform the mutation operation on this chromosome. Otherwise, the system will not perform the mutation operation on this chromosome. In Fig. 1, $T_{initial}$ denotes the initial temperature; T denotes the current system temperature; T_{frozen} denotes the frozen temperature; C and C' denote the current chromosome C and the newly generated chromosome C', respectively; we use the average forecasting rate (AFER) shown in formula (3) as the fitness value of each chromosome in the genetic algorithm for temperature prediction; Δf denotes the difference between the fitness value $fitness(C')$ of the newly generated chromosome C' and the fitness value $fitness(C)$ of the current chromosome C; α denotes the annealing constant. First, the system randomly chooses the ith X gene and the jth Y gene from current chromosome C, and then replaces the value x_i of the ith X gene and the value y_j of the jth Y gene of current chromosome C by the random numbers x_i^* and y_j^* , generated by system, respectively, to form the new generated chromosome C'. Then, the system calculates the fitness value of chromosome C and chromosome C', respectively, and then calculates their difference Δf . If Δf is smaller than or equal to zero, then the newly generated

chromosome C' is always allowed to replace the current chromosome C. Otherwise, the system is allowed to accept the newly generated chromosome C' depending on the probability $e^{(-\Delta f/T)}$. In this situation, if the random number generated by the system is smaller than the probability $e^{(-\Delta f/T)}$, then the newly generated chromosome C' replaces the current chromosome C. Repeatedly perform the above process, until the current system temperature T is smaller than the frozen temperature T_{frozen} . In this case, the simulated annealing mutation process finishes.

```

Procedure SAM ( $T_{initial}, T_{frozen}, \alpha$ )
begin
   $T \leftarrow T_{initial}$ ;
  while ( $T > T_{frozen}$ ) do
    Randomly choose the ith X gene and jth Y gene
    of chromosome C, where  $1 \leq i \leq n, 1 \leq j \leq m$ , n
    is the number of X genes of chromosome C,
    and m is the number of Y gene from
    chromosome C, and replace the value  $x_i$  of the
    ith X gene and the value  $y_j$  of the jth Y gene of
    chromosome C by the random numbers  $x_i^*$  and
     $y_j^*$  generated by the system, respectively, to
    derive newly generated chromosome C' (i.e.,  $C'
    \leftarrow \langle x_1, x_2, \dots, x_i^*, \dots, x_n, y_1, y_2, \dots, y_j^*, \dots,
    y_m \rangle$ );
    Let  $\Delta f \leftarrow fitness(C') - fitness(C)$ ;
     $r \leftarrow$  random number between 0 and 1 generated
    by the system;
    if  $\Delta f \leq 0$  or  $r < e^{(-\Delta f/T)}$  then  $C \leftarrow C'$ ;
     $T \leftarrow T \times \alpha$ ;
  end;
return C
end.

```

Fig. 1. Simulated annealing mutation algorithm.

Step 8: Based on formula (3), calculate the average forecasting error rate (AFER) of each chromosome in the population. If the system has evolved a predefined number of generations, then the chromosome that has the smallest average forecasting error rate is the optimal solution to be used to deal with the forecasting problem; **Stop**. Otherwise, go to Step 5.

We have implemented the proposed method using Visual Basic version 6.0 on a Pentium 4 PC. We use the average forecasting rate (AFER) shown in formula (3) as the fitness value of each chromosome in the genetic algorithm for temperature prediction. We use different annealing constants in the simulated annealing mutation algorithm to forecast the daily average temperature from June 1996 to September 1996 in Taipei, Taiwan, where we partition the universe of discourse U of the main-factor (i.e., the daily average temperature) into 9 intervals and partition the universe of discourse V of the second-factor (i.e., the daily cloud density) into 7 intervals. In other words, each chromosome in a population consists of 8 X genes and 6 Y genes. A comparison of average forecasting error rates of the proposed method with the existing methods is shown in Table 3, where the average forecasting error rates are calculated by executing the proposed method three times, and the number of generations, the population size, the crossover rate, the mutation rate, the initial temperature

and the frozen temperature are 1000, 30, 0.8, 0.05, 100 and 0.0001, respectively. From Table 3, we can see that the proposed method gets smaller forecasting error rates than the methods presented in [2] and [7]. That is, the proposed method gets higher forecasting accuracy rates than the methods presented in [2] and [7] for dealing with temperature prediction.

Table 3. A Comparison of the Average Forecasting Error Rates of the Proposed Method with the Existing Methods

Method	Month	Window Basis								
		w=2	w=3	w=4	w=5	w=6	w=7	w=8		
Chen's Method [2]	June	2.88%	3.16%	3.24%	3.33%	3.39%	3.53%	3.67%		
	July	3.04%	3.76%	4.08%	4.17%	4.35%	4.38%	4.56%		
	August	2.75%	2.77%	3.30%	3.40%	3.18%	3.15%	3.19%		
	September	3.29%	3.10%	3.19%	3.22%	3.39%	3.38%	3.29%		
Lee et al's Method [7]	Month	Order								
		First Order	Second Order	Third Order	Fourth Order	Fifth Order	Sixth Order	Seventh Order	Eighth Order	
	June	1.44%	0.47%	0.50%	0.49%	0.49%	0.50%	0.49%	0.46%	
	July	1.33%	0.46%	0.50%	1.50%	0.50%	0.49%	0.50%	0.50%	
August	1.16%	0.48%	0.48%	0.49%	0.50%	0.49%	0.50%	0.49%		
September	1.28%	0.98%	1.02%	1.12%	1.02%	0.74%	0.86%	0.50%		
The Proposed Method	Annealing Constant α	Month	Order							
			First Order	Second Order	Third Order	Fourth Order	Fifth Order	Sixth Order	Seventh Order	Eighth Order
	0.25	June	0.79%	0.44%	0.42%	0.42%	0.42%	0.44%	0.40%	0.40%
		July	0.66%	0.45%	0.42%	0.41%	0.41%	0.40%	0.41%	0.40%
		August	0.64%	0.43%	0.47%	0.40%	0.41%	0.38%	0.40%	0.45%
		September	0.69%	0.58%	0.59%	0.57%	0.56%	0.57%	0.58%	0.47%
	0.5	June	0.84%	0.50%	0.45%	0.42%	0.38%	0.43%	0.39%	0.46%
		July	0.66%	0.50%	0.47%	0.44%	0.40%	0.38%	0.44%	0.42%
		August	0.69%	0.40%	0.38%	0.37%	0.37%	0.39%	0.42%	0.45%
		September	0.66%	0.62%	0.59%	0.59%	0.56%	0.54%	0.56%	0.53%
	0.9	June	0.79%	0.46%	0.42%	0.44%	0.42%	0.41%	0.46%	0.39%
		July	0.62%	0.46%	0.45%	0.44%	0.44%	0.41%	0.40%	0.40%
August		0.66%	0.40%	0.40%	0.40%	0.36%	0.41%	0.39%	0.44%	
September		0.62%	0.59%	0.61%	0.57%	0.54%	0.59%	0.57%	0.50%	

5 Conclusions

In this paper, we have presented a new method for temperature prediction based on genetic simulated annealing techniques and high-order fuzzy time series, where the simulated annealing techniques are used to deal with the mutation operations of the genetic algorithms. The proposed method uses genetic simulated annealing techniques to adjust the length of each interval in the universe of discourse for temperature prediction to increase the forecasting accuracy rate. From Table 3, we can see that the proposed method gets smaller average forecasting error rates than the methods presented in [2] and [7]. That is, the proposed method gets higher forecasting accuracy rates than the methods presented in [2] and [7].

Acknowledgements

This work was supported in part by the National Science Council, Republic of China, under Grant NSC 93-2213-E-011-018.

References

[1] S. M. Chen, "Forecasting enrollments based on fuzzy time series," *Fuzzy Sets and Systems*, Vol. 81, pp. 311-319, 1996.
 [2] S. M. Chen and J. R. Hwang, "Temperature prediction using fuzzy time series," *IEEE Transactions on Systems, Man, and*

Cybernetics-Part B: Cybernetics, Vol. 30, pp. 263-275, 2000.
 [3] S. M. Chen, "Forecasting enrollments based on high-order fuzzy time series," *Cybernetics and Systems: An International Journal*, Vol. 33, pp. 1-16, 2002.
 [4] J. R. Hwang, S. M. Chen, and C. H. Lee, "Handling forecasting problems using fuzzy time series," *Fuzzy Sets and Systems*, Vol. 100, pp. 217-228, 1998.
 [5] K. Huarng, "Effective lengths of intervals to improve forecasting in fuzzy time series," *Fuzzy Sets and Systems*, Vol. 123, pp. 387-394, 2001.
 [6] K. Huarng, "Heuristic models of fuzzy time series for forecasting," *Fuzzy Sets and Systems*, Vol. 123, pp. 369-386, 2001.
 [7] L. W. Lee and S. M. Chen, "Temperature prediction using genetic algorithms and fuzzy time series," *Proceedings of the 2004 International Conference on Information Management*, Miaoli, Taiwan, Republic of China, pp. 299-306, 2004.
 [8] L. W. Lee, L. H. Wang, S. M. Chen, and Y. H. Leu, "A new method for handling forecasting problems based on two-factors high-order fuzzy time series," *Proceedings of the 2004 Ninth Conference on Artificial Intelligence and Applications*, Taipei, Taiwan, Republic of China, 2004.
 [9] Q. Song and B. S. Chissom, "Some properties of defuzzification neural networks," *Fuzzy Sets and Systems*, Vol. 61, pp. 93-89, 1994.
 [10] Q. Song and B. S. Chissom, "Fuzzy time series and its models," *Fuzzy Sets and Systems*, Vol. 54, pp. 269-277, 1993.
 [11] Q. Song and B. S. Chissom, "Forecasting enrollments with fuzzy time series - Part I," *Fuzzy Sets and Systems*, Vol. 54, pp. 1-9, 1993.
 [12] Q. Song and B. S. Chissom, "Forecasting enrollments with fuzzy time series - Part II," *Fuzzy Sets and Systems*, Vol. 62, pp. 1-8, 1994.
 [13] Q. Song and R. P. Leland, "Adaptive learning defuzzification techniques and applications," *Fuzzy Sets and Systems*, Vol. 81, pp. 321-329, 1996.
 [14] Q. Song, "A note on fuzzy time series model selection with sample autocorrelation functions," *Cybernetics and Systems: An International Journal*, Vol. 34, pp. 93-107, 2003.
 [15] J. Sullivan and W. H. Woodall, "A comparison of fuzzy forecasting and Markov modeling," *Fuzzy Sets and Systems*, Vol. 64, pp. 279-293, 1994.
 [16] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, Vol. 220, pp. 671-680, 1983.
 [17] L. A. Zadeh, "Fuzzy sets," *Information and Control*, Vol. 8, pp. 338-353, 1965.
 [18] J. H. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, Cambridge, MA, 1975.
 [19] M. Gen and R. Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, 1997.
 [20] D. E. Goldberg, *Genetic Algorithm in Search, Optimization, and Machine Learning*, Addison-Wesley, MA, 1989.
 [21] Central Weather Bureau, *The Historical Data of the Daily Average Temperature and Daily Cloud Density* (from January 1995 to September 1996), Taipei, Taiwan, R. O. C., 1996.
 [22] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 22, pp. 1414-1427, 1992.