Expert Systems with Applications 37 (2010) 2756–2770

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Temperature prediction and TAIFEX forecasting based on fuzzy relationships and MTPSO techniques ☆

Ling-Yuan Hsu [a,b], Shi-Jinn Horng [a,c,d,f,*], Tzong-Wann Kao [e], Yuan-Hsin Chen [d], Ray-Shine Run [d], Rong-Jian Chen [d], Jui-Lin Lai [d], I-Hong Kuo [b]

[a] Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, 106 Taipei, Taiwan
[b] Department of Information Management, ST. Mary's Medicine Nursing and Management College, I-Lan, Taiwan
[c] Department of Electrical Engineering, National Taiwan University of Science and Technology, 106 Taipei, Taiwan
[d] Department of Electronic Engineering, National United University, 36003 Miao-Li, Taiwan
[e] Department of Electronic Engineering, Technology and Science Institute of Northern Taiwan, Taipei, Taiwan
[f] Department of Computer Science, Georgia State University, USA

ARTICLE INFO

Keywords:
Forecasting
Two-factors
Fuzzy relationships
Fuzzy time series
Modify turbulent particle swarm optimization

ABSTRACT

In this paper, we proposed a modified turbulent particle swarm optimization (named MTPSO) method for the temperature prediction and the Taiwan Futures Exchange (TAIFEX) forecasting, based on the two-factor fuzzy time series and particle swarm optimization. The MTPSO model can be dealt with two main factors easily and accurately, which are the lengths of intervals and the content of forecast rules. The experimental results of the temperature prediction and the TAIFEX forecasting show that the proposed model is better than any existing models and it can get better quality solutions based on the high-order fuzzy time series, respectively.

1. Introduction

Because nobody knows everything till tomorrow, there is always a hope for the future of what happens to be unpredictable. For example, the stock market forecasts for tomorrow, the weather forecasts for tomorrow, or the enrollments forecast for the next year. Forecast is a very interesting research topic; it attracted many researchers over the past few decades. They were using some forecasting techniques to forecast the tread of stock market (Cheng, Chen, Teoh, & Chiang, 2008; Chu, Chen, Cheng, & Huang, 2009; Huarng & Yu, 2005; Lee, Wang, Chen, & Leu, 2006; Lee, Wang, & Chen, 2007, 2008; Yu & Huarng, 2008; Wang & Chen, 2009; Huarng, 2001a, 2001b), tomorrow's temperature (Chen & Hwang, 2000; Lee et al., 2006; Lee, Wang, & Chen, 2007, 2008; Wang & Chen, 2009; Li, Chen, & Li, 1988), enrollments of the next year (Chen, 1996; Cheng et al., 2008; Huarng, 2001a, 2001b; Kuo et al., 2009; Singh, 2009; Singh, 2007a, 2007b; Song & Chissom, 1993b, 1994; Tsai & Wu et al., 2000), and etc. As we know, to forecast these matters is generally believed to be a very difficult task. It looks like the performance of a random walk process on a different time. Obviously, we need to investigate some intelligent forecasting paradigms to solve the forecasting problems.

Zadeh (1965) proposed the fuzzy set theory first and then got fruitful achievements both in theory and applications. In Li et al. (1988) proposed a method for the weather forecast considering fuzziness between the demarcation lines of fuzzy grades and the membership functions of fuzzy grade. Song and Chissom introduced a new forecast model based on the concept of fuzzy time series (Song & Chissom, 1993a, 1993b, 1994). They use the time-variant fuzzy time series model and the time-invariant fuzzy time series model based on the fuzzy set theory for forecasting the enrollments of the University of Alabama. Chen improved the fuzzy time series model by max–min composition operations (Chen, 1996). Huarng presented a method to improve forecasting results in forecasting the enrollments of the University of Alabama and the Taiwan Futures Exchange (TAIFEX) (Huarng, 2001a, 2001b). Chen and Hwang presented a method for the temperature prediction based on fuzzy time series (Chen & Hwang, 2000). Lee et al. presented methods for forecasting the temperature and the TAIFEX based on two-factors high-order fuzzy time series (Lee et al., 2007, 2008). They also use the genetic algorithm and genetic simulated annealing in it. Kuo et al. presented an improved method for forecasting enrollments based on the fuzzy time series and particle

☆ This work was supported in part by the National Science Council under Contract Nos. NSC 97-2221-E-239-022-, 98-2221-E-011-133-MY3, 98-2923-E-011-004-MY3.

* Corresponding author. Address: Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, 106 Taipei, Taiwan. Tel.: +886 2 27376700; fax: +886 2 27301081.

E-mail addresses: D9415008@mail.ntust.edu.tw (L.-Y. Hsu), horngsj@yahoo.com.tw (S.-J. Horng), tkao@ms6.hinet.net (T.-W. Kao), yschen@nuu.edu.tw (Y.-H. Chen), run5116@ms16.hinet.net (R.-S. Run), rjchen@nuu.edu.tw (R.-J. Chen), jllai@nuu.edu.tw (J.-L. Lai), yihonguo@gmail.com (I-H. Kuo).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.09.015

swarm optimization (Kuo et al., 2009). They propose a more effective and accuracy method to forecasting enrollments.

For these forecast methods, there are two main interesting factors affecting the forecast accuracy, which are the lengths of intervals and the content of forecast rules. By considering these two main factors, in this paper, we proposed a new method for the temperature prediction and the TAIFEX forecasting, based on two-factor high-order fuzzy relationships and particle swarm optimization. The proposed method uses "modified turbulent particle swarm optimization" (named MTPSO) techniques to find the proper content of two main factors to improve the forecasted accuracy. The experimental results show that the new model is more precise than the existing methods.

The remainder of this paper is organized as follows: Section 2 briefly overviews the procedure of temperature prediction using the fuzzy time series. Section 3 describes the particle swarm optimization (PSO). Section 4 discusses the details of the new proposed forecast model. Section 5 discusses the experimental results obtained from the new proposed forecast model. Finally, Section 6 summarizes the contribution of this paper and conclusions.

2. The procedure of the temperature prediction using the fuzzy time series

In this section, a brief overview of the fuzzy time series is included within the forecasting procedure in the temperature prediction. The procedure also applies to forecast TAIFEX. The concept of the fuzzy time series was introduced by Song and Chissom (1993a, 1993b, 1994) based on the fuzzy set theory (Zadeh, 1965). It can deal with the forecasting problem where the historical data are linguistic values. The procedure of the temperature prediction using the fuzzy time series is described as follows:

Let U be the universe of discourse, where $U = \{u_1, u_2, \ldots, u_m\}$. A fuzzy set A_i of U is defined by

$$A_i = u_1/f_{A_i}(u_1) + u_2/f_{A_i}(u_2) + \cdots + u_m/f_{A_i}(u_m), \tag{1}$$

where f_{A_i} is the membership function of the fuzzy set $A_i, f_{A_i} : U \to [0,1]$. u_j is an element of fuzzy set A_i, and $f_{A_i}(u_j)$ is the degree of membership of u_j belonging to A_i. $f_{A_i}(u_j) \in [0,1]$ and $1 \leqslant j \leqslant m$.

Let $Y(t)(t = \ldots, 0, 1, 2, \ldots)$ be the universe of discourse on which the fuzzy sets $f_i(t)(i = 1, 2, \ldots)$ are defined and $F(t)$ is a collection of $f_1(t), f_2(t)$, Then $F(t)$ is called a fuzzy time series defined on $Y(t)(t = \ldots, 0, 1, 2, \ldots)$.

Suppose $F(t)$ is caused by $F(t-1)$ and it is denoted by the fuzzy relationship $F(t-1) \to F(t)$. Then this relation can be expressed as $F(t) = F(t-1) \circ R(t, t-1)$, where "$\circ$" is usually a max–min operator, $R(t, t-1)$ is the union of all fuzzy relations and each of $R(t, t-1)$ is a fuzzy relationship between $F(t-1)$ and $F(t)$. $F(t) = F(t-1) \circ R(t, t-1)$ is called the first-order model of $F(t)$.

Suppose $F(t)$ is a fuzzy time series and $R(t, t-1)$ is a first-order model of $F(t)$. If $R(t, t-1) = R(t-1, t-2)$ for any time $t, R(t, t-1)$ is independent of t, then $F(t)$ is called a time-invariant fuzzy time series.

Let $F(t)$ be a fuzzy time series. If $F(t)$ is caused by $F(t-1)$, $F(t-2), \ldots$ and $F(t-\lambda)$, then the λth-order fuzzy relationship is represented by $F(t-\lambda), \ldots, F(t-2), F(t-1) \to F(t)$, where $F(t-\lambda), \ldots, F(t-2), F(t-1)$ is the current state and $F(t)$ is the next state.

Let $F_A(t)$ and $F_B(t)$ be two fuzzy time series. If $F_A(t)$ is caused by $(F_A(t-1), F_B(t-1)), (F_A(t-2), F_B(t-2)), \ldots, (F_A(t-\lambda), F_B(t-\lambda))$, then the two-factor λth-order fuzzy relationship is represented by $(F_A(t-\lambda), F_B(t-\lambda)), \ldots, (F_A(t-2), F_B(t-2)), (F_A(t-1), F_B(t-1)) \to F_A(t)$, where $F_A(t)$ and $F_B(t)$ $(t = \ldots 0, 1, 2, \ldots)$ are called the main-factor fuzzy time series and the second-factor fuzzy time series, respectively. $(F_A(t-\lambda), F_B(t-\lambda)), \ldots, (F_A(t-2), F_B(t-2))$, $(F_A(t-1), F_B(t-1))$ and $F_A(t)$ are called the current state and the next state of the fuzzy relationship $(F_A(t-\lambda), F_B(t-\lambda)), \ldots, (F_A(t-2), F_B(t-2)), (F_A(t-1), F_B(t-1)) \to F_A(t)$, respectively.

Two-factor fuzzy time series can be used to predict temperature which is described as follows:

2.1. Step 1: Define two universes of discourse $Y_A(t)$ and $Y_B(t)$

Let $Y_A(t)$ and $Y_B(t)$ be two historical data on day t(June-1-1996 $\leqslant t \leqslant$ September-30-1996). For defining the universe, first find the minimum data D_{min} and the maximum data D_{max} of known historical data. Based on D_{min} and D_{max}, define the universe $Y_A(t)$ (or $Y_B(t)$) as $[D_{min} - D_1, D_{max} + D_2]$ where D_1 and D_2 denote the buffers to adjust the lower bound and the upper bound of the universe of discourse, $Y_A(t)$ (or $Y_B(t)$), respectively.

According to Table 1, it is obvious that the daily minimum temperature and maximum temperature are $D_{min} = 23.3$ °C and $D_{max} = 31.6$ °C, respectively. For convenience of illustrating the forecasting example here, we set $D_1 = 0.3$ °C and $D_2 = 0.4$ °C. We called it the main-factor of the fuzzy time series and get the universe of discourse on $Y_A(t) = [23.0, 32.0]$.

According to Table 2, it is obvious that the daily minimum cloud density and maximum cloud density are $D_{min} = 3$ and $D_{max} = 100$, respectively. As for the illustration, we set $D_1 = 3$ and $D_2 = 0$. We called it the second-factor of the fuzzy time series and get the universe of discourse on $Y_B(t) = [0.0, 100.0]$.

2.2. Step 2: Partition the universes $Y_A(t)$ and $Y_B(t)$ into several intervals

After *Step 1*, the universes of discourse on $Y_A(t)$ and $Y_B(t)$ are then cut into the pre-defined number of intervals. First, we divide $Y_A(t)$ into 9 intervals with equal lengths, and use u_1, u_2, u_3,

Table 1
Historical data of the daily average temperature from June 1996 to September 1996 in Taipei, Taiwan (Unit: °C) (Taiwan Central Weather Bureau, 1996).

Days	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	

u_4, u_5, u_6, u_7, u_8 and u_9 to denote each interval, respectively. Thus, the nine intervals are $u_1 = [23.0, 24.0)$, $u_2 = [24.0, 25.0)$, $u_3 = [25.0, 26.0)$, $u_4 = [26.0, 27.0)$, $u_5 = [27.0, 28.0)$, $u_6 = [28.0, 29.0)$, $u_7 = [29.0, 30.0)$, $u_8 = [30.0, 31.0)$ and $u_9 = [31.0, 32.0]$. Then calculate the midpoint of intervals, which are $m_1 = 23.5, m_2 = 24.5, m_3 = 25.5, m_4 = 26.5, m_5 = 27.5$, $m_6 = 28.5$, $m_7 = 29.5, m_8 = 30.5$ and $m_9 = 31.5$, respectively. Second, we divide $Y_B(t)$ into 7 intervals with equal lengths, and use $v_1, v_2, v_3, v_4, v_5, v_6$ and v_7 to denote each interval, respectively. Thus, the seven intervals are $v_1 = [0.0, 14.29)$, $v_2 = [14.29, 28.57)$, $v_3 = [28.57, 42.86)$, $v_4 = [42.86, 57.14)$, $v_5 = [57.14, 71.43)$, $v_6 = [71.43, 85.71)$ and $v_7 = [85.71, 100.0]$.

2.3. Step 3: Define fuzzy set linguistic terms

According to the number of intervals as mentioned above, let "historical data" be the linguistic variable. First, define the main-factor represented by fuzzy sets A_i. Each $A_i (1 \leqslant i \leqslant 9)$ denotes a fuzzy set, and its definition is described in Eq. (1). In Eq. (2), the symbol $f_{A_i}(u_j) \in [0, 1] (1 \leqslant j \leqslant 9)$ is a real number and denotes the membership degree that u_j belongs to $A_i (1 \leqslant i \leqslant 9)$. In other words, A_i denotes a fuzzy set $\{u_1, u_2, \ldots, u_9\}$ with different membership degree $\{f_{A_i}(u_1), f_{A_i}(u_2), \ldots, f_{A_i}(u_9)\}$. The detailed definitions of all main-factor fuzzy sets are described in Eq. (2).

$$\begin{aligned}
A_1 &= u_1/1 + u_2/0.5 + u_3/0 + u_4/0 + u_5/0 + u_6/0 + u_7/0 + u_8/0 + u_9/0,\\
A_2 &= u_1/0.5 + u_2/1 + u_3/0.5 + u_4/0 + u_5/0 + u_6/0 + u_7/0 + u_8/0 + u_9/0,\\
A_3 &= u_1/0 + u_2/0.5 + u_3/1 + u_4/0.5 + u_5/0 + u_6/0 + u_7/0 + u_8/0 + u_9/0,\\
A_4 &= u_1/0 + u_2/0 + u_3/0.5 + u_4/1 + u_5/0.5 + u_6/0 + u_7/0 + u_8/0 + u_9/0,\\
A_5 &= u_1/0 + u_2/0 + u_3/0 + u_4/0.5 + u_5/1 + u_6/0.5 + u_7/0 + u_8/0 + u_9/0,\\
A_6 &= u_1/0 + u_2/0 + u_3/0 + u_4/0 + u_5/0.5 + u_6/1 + u_7/0.5 + u_8/0 + u_9/0,\\
A_7 &= u_1/0 + u_2/0 + u_3/0 + u_4/0 + u_5/0 + u_6/0.5 + u_7/1 + u_8/0.5 + u_9/0,\\
A_8 &= u_1/0 + u_2/0 + u_3/0 + u_4/0 + u_5/0 + u_6/0 + u_7/0.5 + u_8/1 + u_9/0.5,\\
A_9 &= u_1/0 + u_2/0 + u_3/0 + u_4/0 + u_5/0 + u_6/0 + u_7/0 + u_8/0.5 + u_9/1.
\end{aligned} \quad (2)$$

Then define the second-factor represented by fuzzy sets B_i. Each $B_i (1 \leqslant i \leqslant 7)$ denotes a fuzzy set, and its definition is described similarly to A_i in Eq. (1). In Eq. (3), the symbol $f_{B_i}(v_j) \in [0, 1] (1 \leqslant j \leqslant 7)$ is a real number and denotes the membership degree that v_j belongs to $B_i (1 \leqslant i \leqslant 7)$. In other words, B_i denotes a fuzzy set $\{v_1, v_2, \ldots, v_7\}$ with different membership degree $\{f_{B_i}(v_1), f_{B_i}(v_2), \ldots, f_{B_i}(v_7)\}$. The detailed definitions of all second-factor fuzzy sets are described in Eq. (3).

$$\begin{aligned}
B_1 &= v_1/1 + v_2/0.5 + v_3/0 + v_4/0 + v_5/0 + v_6/0 + v_7/0,\\
B_2 &= v_1/0.5 + v_2/1 + v_3/0.5 + v_4/0 + v_5/0 + v_6/0 + v_7/0,\\
B_3 &= v_1/0 + v_2/0.5 + v_3/1 + v_4/0.5 + v_5/0 + v_6/0 + v_7/0,\\
B_4 &= v_1/0 + v_2/0 + v_3/0.5 + v_4/1 + v_5/0.5 + v_6/0 + v_7/0,\\
B_5 &= v_1/0 + v_2/0 + v_3/0 + v_4/0.5 + v_5/1 + v_6/0.5 + v_7/0,\\
B_6 &= v_1/0 + v_2/0 + v_3/0 + v_4/0 + v_5/0.5 + v_6/1 + v_7/0.5,\\
B_7 &= v_1/0 + v_2/0 + v_3/0 + v_4/0 + v_5/0 + v_6/0.5 + v_7/1.
\end{aligned} \quad (3)$$

2.4. Step 4: Fuzzify all historical data

In order to find out an equivalent fuzzy set using Eqs. (2) and (3) to all historical data. The way to fuzzify a historical data is to find an interval to which it belongs and assign the corresponding linguistic value to it. For example, the historical data on August 1, 1996, the actual daily average temperature and daily cloud density are 27.1 °C and 100%, respectively and they belong to interval $u_5 = [27.0, 28.0)$ and interval $v_7 = [85.71, 100.0]$, respectively. Hence, we assign the fuzzy set A_5 corresponding to interval u_5 of the main-factor and assign the fuzzy set B_7 corresponding to interval v_7 of the second-factor, respectively. According to Tables 1 and 2, the results of fuzzification are listed in Table 3 by using Eqs. (2) and (3) to fuzzify the historical data of the daily average temperature and the daily cloud density.

Table 2
Historical data of the daily cloud density from June 1996 to September 1996 in Taipei, Taiwan (Unit: %) (Taiwan Central Weather Bureau, 1996).

Days	Months			
	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	

Let $Y_A(t)$ and $Y_B(t)$ be two historical data time series on day t. The purpose of *Step 4* is to get two fuzzy time series $F_A(t)$ and $F_B(t)$ on $Y_A(t)$ and $Y_B(t)$. Each element of $Y_A(t)$ and $Y_B(t)$ is a real number with respect to the actual daily average temperature and the actual daily cloud density, respectively. But each element of $F_A(t)$ and $F_B(t)$ is a fuzzy set with respect to the corresponding element of $Y_A(t)$ and $Y_B(t)$, respectively. For example, in Table 3, Y_A(August-1-1996) = 27.1 and F_A(August-1-1996) = A_5, Y_B(August-1-1996) = 100 and F_B(August-1-1996) = B_7, Y_A(August-7-1996) = 29.0 and F_A(August-7-1996) = A_7, Y_B(August-7-1996) = 71 and F_B(August-7-1996) = B_5, and so on.

2.5. Step 5: Construct all two-factor λth-order fuzzy relationship groups

After two fuzzy time series $F_A(t)$ and $F_B(t)$ have been created, we can find out all fuzzy relationships under different orders. The way to construct all two-factor first-order fuzzy relationship is to find any relationship consisting of the type $(F_A(t-1), F_B(t-1)) \rightarrow F_A(t)$, where $F_A(t-1), F_B(t-1)$, and $F_A(t)$ are called the current state and the next state, respectively. Then a fuzzy relationship can be obtained by replacing $F_A(t-1), F_B(t-1)$, and $F_A(t)$ with the corresponding fuzzy set. For example, (F_A(August-1-1996), F_B(August-1-1996)) → F_A(August-2-1996) is a relationship, and a fuzzy relationship $(A_5, B_7) \rightarrow A_6$ is obtained by replacing (F_A(August-1-1996), F_B(August-1-1996)) and F_A(August-2-1996) to (A_5, B_7) and A_6, respectively. Another fuzzy relationship $(A_6, B_5) \rightarrow A_7$ is got as (F_A(August-3-1996), F_B(August-3-1996)) and F_A(August-4-1996).

Table 3
Fuzzified the daily average temperature and the daily cloud density from August-1-1996 to August-31-1996 in Taipei, Taiwan.

Days	Actual daily average temperature (°C)	Fuzzy sets $F_A(t)$	Actual daily cloud density (%)	Fuzzy sets $F_B(t)$
August-1-1996	27.1	A_5	100	B_7
August-2-1996	28.9	A_6	78	B_6
August-3-1996	28.9	A_6	68	B_5
August-4-1996	29.3	A_7	44	B_4
August-5-1996	28.8	A_6	56	B_4
August-6-1996	28.7	A_6	89	B_7
August-7-1996	29.0	A_7	71	B_5
August-8-1996	28.2	A_6	28	B_2
August-9-1996	27.0	A_5	70	B_5
August-10-1996	28.3	A_6	44	B_4
August-11-1996	28.9	A_6	48	B_4
August-12-1996	28.1	A_6	76	B_6
August-13-1996	29.9	A_7	50	B_4
August-14-1996	27.6	A_5	84	B_6
August-15-1996	26.8	A_4	69	B_5
August-16-1996	27.6	A_5	78	B_6
August-17-1996	27.9	A_5	39	B_3
August-18-1996	29.0	A_7	20	B_2
August-19-1996	29.2	A_7	24	B_2
August-20-1996	29.8	A_7	25	B_2
August-21-1996	29.6	A_7	19	B_2
August-22-1996	29.3	A_7	46	B_4
August-23-1996	28.0	A_6	41	B_3
August-24-1996	28.3	A_6	34	B_3
August-25-1996	28.6	A_6	29	B_3
August-26-1996	28.7	A_6	31	B_3
August-27-1996	29.0	A_7	41	B_3
August-28-1996	27.7	A_5	14	B_1
August-29-1996	26.2	A_4	28	B_2
August-30-1996	26.0	A_4	33	B_3
August-31-1996	27.7	A_5	26	B_2

In order to construct all two-factor λth-order ($\lambda \geqslant 2$) fuzzy relationships, it's necessary to find any relationship consisting of the type $(F_A(t-\lambda), F_B(t-\lambda)),\ldots,(F_A(t-2), F_B(t-2)), (F_A(t-1), F_B(t-1)) \rightarrow F_A(t)$, where $(F_A(t-\lambda), F_B(t-\lambda)),\ldots,(F_A(t-2), F_B(t-2)), (F_A(t-1), F_B(t-1))$ and $F_A(t)$ are called the current state and the next state, respectively. Then a λth-order fuzzy relationship is obtained by replacing the corresponding fuzzy set. For example, suppose $\lambda = 2$, a fuzzy relationship $(A_5,B_7), (A_6,B_6) \rightarrow A_6$ is obtained from (F_A(August-1-1996), F_B(August-1-1996)),(F_A(August-2-1996), F_B(August-2-1996)) $\rightarrow F_A$ (August-3-1996). According to Table 3, the complete two-factor second-order fuzzy relationships are listed in Table 4, where there are 26 groups and each group of two-factor second-order fuzzy relationship is fuzzified from the historical data of the daily average temperature and the daily cloud density ranged from August-1-1996 to August-31-1996.

In Table 4, group 26 consists of the fuzzy relationship $(A_4,B_3), (A_5,B_2) \rightarrow \#$ as it is created by the relationship (F_A(August-30-1996), F_B(August-30-1996)), (F_A(August-31-1996), F_B(August-31-1996)) $\rightarrow F_A$(September-1-1996), since the linguistic value of F_A(September-1-1996) is unknown within the historical data from August-1-1996 to August-31-1996. Hence, we use symbol "#" to denote the unknown value (For the sake of illustration, here we assume that we do not have the historical data of temperature and cloud density on September-1-1996.).

2.6. Step 6: Calculate the forecasting value and create all fuzzy forecast rules based on all fuzzy relationship groups

In this step, it is necessary to calculate the forecasting value and create all fuzzy forecast rules based on all two-factor λth-order fuzzy relationship groups, respectively, and then find out the matched forecast rule to get the forecast value.

Table 4
Two-factor second-order fuzzy relationship groups.

Days	Fuzzy relationships	Group ID
August-3-1996	$(A_5,B_7), (A_6,B_6) \rightarrow A_6$	1
August-4-1996	$(A_6,B_6), (A_6,B_5) \rightarrow A_7$	2
August-5-1996	$(A_6,B_5), (A_7,B_4) \rightarrow A_6$	3
August-6-1996	$(A_7,B_4), (A_6,B_4) \rightarrow A_6$	4
August-7-1996	$(A_6,B_4), (A_6,B_7) \rightarrow A_7$	5
August-8-1996	$(A_6,B_7), (A_7,B_5) \rightarrow A_6$	6
August-9-1996	$(A_7,B_5), (A_6,B_2) \rightarrow A_5$	7
August-10-1996	$(A_6,B_2), (A_5,B_5) \rightarrow A_6$	8
August-11-1996	$(A_5,B_5), (A_6,B_4) \rightarrow A_6$	9
August-12-1996	$(A_6,B_4), (A_6,B_4) \rightarrow A_6$	10
August-13-1996	$(A_6,B_4), (A_6,B_6) \rightarrow A_7$	11
August-14-1996	$(A_6,B_6), (A_7,B_4) \rightarrow A_5$	12
August-15-1996	$(A_7,B_4), (A_5,B_6) \rightarrow A_4$	13
August-16-1996	$(A_5,B_6), (A_4,B_5) \rightarrow A_5$	14
August-17-1996	$(A_4,B_5), (A_5,B_6) \rightarrow A_5$	15
August-18-1996	$(A_5,B_6), (A_5,B_3) \rightarrow A_7$	16
August-19-1996	$(A_5,B_3), (A_7,B_2) \rightarrow A_7$	17
August-20-1996	$(A_7,B_2), (A_7,B_2) \rightarrow A_7$	18
August-21-1996	$(A_7,B_2), (A_7,B_2) \rightarrow A_7$	18
August-22-1996	$(A_7,B_2), (A_7,B_2) \rightarrow A_7$	18
August-23-1996	$(A_7,B_2), (A_7,B_4) \rightarrow A_6$	19
August-24-1996	$(A_7,B_4), (A_6,B_3) \rightarrow A_6$	20
August-25-1996	$(A_6,B_3), (A_6,B_3) \rightarrow A_6$	21
August-26-1996	$(A_6,B_3), (A_6,B_3) \rightarrow A_6$	21
August-27-1996	$(A_6,B_3), (A_6,B_3) \rightarrow A_7$	21
August-28-1996	$(A_6,B_3), (A_7,B_3) \rightarrow A_5$	22
August-29-1996	$(A_7,B_3), (A_5,B_1) \rightarrow A_4$	23
August-30-1996	$(A_5,B_1), (A_4,B_2) \rightarrow A_4$	24
August-31-1996	$(A_4,B_2), (A_4,B_3) \rightarrow A_5$	25
September-1-1996	$(A_4,B_3), (A_5,B_2) \rightarrow \#$	26

Suppose the two-factor λth-order fuzzified historical data before day i are $(A_{i\lambda},B_{i\lambda}),\ldots,(A_{i2},B_{i2})$ and (A_{i1},B_{i1}), where $A_{i\lambda},\ldots,A_{i2}$ and A_{i1} are fuzzified values of the main-factor of day $i-\lambda,\ldots,i-2$, and $i-1$, respectively, $B_{i\lambda},\ldots,B_{i2}$ and B_{i1} are fuzzified values of the second-factor of day $i-\lambda,\ldots,i-2$, and $i-1$, respectively, and $\lambda \geqslant 2$.

First, we calculate the forecasting value using the two-factor λth-order fuzzy relationship groups based on the following cases.

Case 1: one member only
Suppose there is a group which has a member having the two-factor λth-order fuzzy relationship shown as follows:

$$(A_{i\lambda}, B_{i\lambda}), \ldots, (A_{i2}, B_{i2}), (A_{i1}, B_{i1}) \rightarrow A_j,$$

where the maximum membership value of A_j occurs at interval u_j, and the midpoint of interval u_j is m_j, then the forecasting value of day i is m_j. For example, the group 1 in Table 4. The fuzzy relationship is $(A_5,B_7), (A_6,B_6) \rightarrow A_6$, and the midpoint m_6 of A_6 is 28.5. Then the forecast value of day i(i.e. August-3-1996) is 28.5.
Case 2: multiple members
Suppose there is a group which has multiple members and each member has a two-factor λth-order fuzzy relationship shown as follows:

$$(A_{i\lambda}, B_{i\lambda}), \ldots, (A_{i2}, B_{i2}), (A_{i1}, B_{i1}) \rightarrow A_{j1},$$
$$(A_{i\lambda}, B_{i\lambda}), \ldots, (A_{i2}, B_{i2}), (A_{i1}, B_{i1}) \rightarrow A_{j2},$$
$$\vdots$$
$$(A_{i\lambda}, B_{i\lambda}), \ldots, (A_{i2}, B_{i2}), (A_{i1}, B_{i1}) \rightarrow A_{jp},$$

where the maximum membership values of $A_{j1},A_{j2},\ldots$ and A_{jp} occur at intervals $u_{j1},u_{j2},\ldots$, and u_{jp}, respectively, and the midpoints of $u_{j1},u_{j2},\ldots$ and u_{jp} are $m_{j1},m_{j2},\ldots$ and m_{jp}, respectively, then the forecasting value of day i is the average of $m_{j1},m_{j2},\ldots$ and m_{jp}. For example, the group 21 in Table 4, the midpoint m_6 of A_6 for the first member is 28.5, and the midpoint m_7 of A_7 for the second member is 29.5. Then the forecast value is $(28.5 + 29.5)/2 = 29.0$.

Case 3: a member with an unknown value
Suppose there is a group which has a member having the two-factor λth-order fuzzy relationship with an unknown value shown as follows:

$(A_{i\lambda}, B_{i\lambda}), \ldots, (A_{i2}, B_{i2}), (A_{i1}, B_{i1}) \rightarrow \#,$

where the symbol "#" denotes an unknown value, the maximum membership values of $A_{i\lambda}, A_{i(\lambda-1)}, \ldots$ and A_{i1} occur at intervals $u_{i\lambda}, u_{i(\lambda-1)}, \ldots$ and u_{i1}, respectively, and the midpoints of $u_{i\lambda}, u_{i(\lambda-1)}, \ldots$ and u_{i1} are $m_{i\lambda}, m_{i(\lambda-1)}, \ldots$ and m_{i1}, respectively, then the forecasting value of day i is calculated as follows:

$$m_{i1} + \sum_{j=2}^{\lambda} \frac{(m_{i(j-1)} - m_{ij})}{2^{(j-1)}}. \quad (4)$$

In this paper, we use Eq. (4) to forecast the unknown value. The method of Eq. (4) is based on the latest past linguistic value, plus the half of the difference between one day earlier and two days earlier, plus the quarter of the difference between two days earlier and three days earlier and so on. For example, the group 26 in Table 4, the fuzzy relationship is $(A_4, B_3), (A_5, B_2) \rightarrow \#$, and the midpoint m_5 of A_5 is 27.5, and the midpoint m_4 of A_4 is 26.5. Then the forecast value is 27.5 + (27.5–26.5)/2 = 28.0.

The complete forecasted values for all groups in Table 4 are listed in Table 5, where symbol $m_j (1 \leqslant j \leqslant 9)$ denotes the midpoint of interval u_j $(1 \leqslant j \leqslant 9)$. Now we are going to create all fuzzy forecast rules. A forecast rule consists of two parts, the matching part filled up with the current state of the fuzzy relationships of the same group, and the forecasting part filled up with the above mentioned forecasted value. Since all fuzzy relationships of the same group have the same current state, it is reasonable to apply the current state in the matching part of a fuzzy forecast rule. The complete two-factor second-order fuzzy forecast rules created based on Table 5 are listed in Table 6. How to forecast the next training or testing data $Y_A(t)$ based on all forecast rules is described in the next step.

Table 5
The complete forecasted values for all groups of the two-factors second-order fuzzy relationships.

Group ID	Fuzzy relationships	Forecasted values
1	$(A_5,B_7), (A_6,B_6) \rightarrow A_6$	28.5 (=m_6)
2	$(A_6,B_6), (A_6,B_5) \rightarrow A_7$	29.5 (=m_7)
3	$(A_6,B_5), (A_7,B_4) \rightarrow A_6$	28.5 (=m_6)
4	$(A_7,B_4), (A_6,B_4) \rightarrow A_6$	28.5 (=m_6)
5	$(A_6,B_4), (A_6,B_7) \rightarrow A_7$	29.5 (=m_7)
6	$(A_6,B_7), (A_7,B_5) \rightarrow A_6$	28.5 (=m_6)
7	$(A_7,B_5), (A_6,B_2) \rightarrow A_5$	27.5 (=m_5)
8	$(A_6,B_2), (A_5,B_5) \rightarrow A_6$	28.5 (=m_6)
9	$(A_5,B_5), (A_6,B_4) \rightarrow A_6$	28.5 (=m_6)
10	$(A_6,B_4), (A_6,B_4) \rightarrow A_6$	28.5 (=m_6)
11	$(A_6,B_4), (A_6,B_6) \rightarrow A_7$	29.5 (=m_7)
12	$(A_6,B_6), (A_7,B_4) \rightarrow A_5$	27.5 (=m_5)
13	$(A_7,B_4), (A_5,B_6) \rightarrow A_4$	26.5 (=m_4)
14	$(A_5,B_6), (A_4,B_5) \rightarrow A_5$	27.5 (=m_5)
15	$(A_4,B_5), (A_5,B_6) \rightarrow A_5$	27.5 (=m_5)
16	$(A_5,B_6), (A_5,B_3) \rightarrow A_7$	29.5 (=m_7)
17	$(A_5,B_3), (A_7,B_2) \rightarrow A_7$	29.5 (=m_7)
18	$(A_7,B_2), (A_7,B_2) \rightarrow A_7$	29.5 (=m_7)
19	$(A_7,B_2), (A_7,B_4) \rightarrow A_6$	28.5 (=m_6)
20	$(A_7,B_4), (A_6,B_3) \rightarrow A_6$	28.5 (=m_6)
21	$(A_6,B_3), (A_6,B_3) \rightarrow A_6$, $(A_6,B_3), (A_6,B_3) \rightarrow A_7$	29.0 (=($m_6 + m_7$)/ 2)
22	$(A_6,B_3), (A_7,B_3) \rightarrow A_5$	27.5 (=m_5)
23	$(A_7,B_3), (A_5,B_1) \rightarrow A_4$	26.5 (=m_4)
24	$(A_5,B_1), (A_4,B_2) \rightarrow A_4$	26.5 (=m_4)
25	$(A_4,B_2), (A_4,B_3) \rightarrow A_5$	27.5 (=m_5)
26	$(A_4,B_3), (A_5,B_2) \rightarrow \#$	28.0 (calculated by case 3 of **Step 6**)

2.7. Step 7: Forecast the training or the testing data based on the forecast rules

When we are going to forecast the data $Y_A(t)$ based on the forecast rules, it is necessary to find out the matched forecast rule to get the forecasted value. If we use the two-factor second-order forecast rules in Table 6 to forecast the data $Y_A(t)$, we need to find out the corresponding linguistic values of $(F_A(t-2), F_B(t-2))$ and $(F_A(t-1), F_B(t-1))$ with respect to the data $(Y_A(t-2), Y_B(t-2))$ and $(Y_A(t-2), Y_B(t-2))$, and compare with all matching parts of the forecast rules, then we can get a forecasted value from the forecasting part of the matched forecast rule. For example, if we want to forecast the data Y_A(August-3-1996), it is necessary to find out the corresponding linguistic value of (F_A(August-1-1996), F_B(August-1-1996)) (i.e. (A_5,B_7)) and (F_A (August-2-1996), F_B(August-2-1996)) (i.e. (A_6,B_6)) with respect to (Y_A(August-1-1996), Y_B(August-1-1996)) and (Y_A(August-2-1996), Y_B(August-2-1996)) in Table 3 and get the following pattern:

"if (F_A(August-1-1996), F_B(August-1-1996)) equals (A_5,B_7), and (F_A(August-2-1996), F_B(August-2-1996)) equals (A_6,B_6) then forecast Y_A (August-3-1996) = ?"

If we replace the values August-1-1996, August-2-1996 and August-3-1996 in the above pattern with the symbols $t-2$, $t-1$, and t, respectively, then the above pattern equals the following pattern:

"if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_7), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_6) then forecast $Y_A(t)$ = ?"

After using the rule 1 listed in Table 6, a forecasted value (i.e. 28.5) is got for the forecasted value for the above pattern.

As another example, if we want to forecast the data Y_A(September-1-1996), it is necessary to find out the corresponding linguistic value of (F_A(August-30-1996), F_B(August-30-1996)) (i.e. (A_4,B_3)) and (F_A(August-31-1996), F_B(August-31-1996)) (i.e. (A_5,B_2)) with respect to (Y_A(August-30-1996), Y_B(August-30-1996)) and (Y_A(August-31-1996), Y_B(August-31-1996)) in Table 3 and get the following pattern:

"if (F_A(August-30-1996), F_B(August-30-1996)) equals (A_4,B_3), and (F_A(August-31-1996), F_B(August-31-1996)) equals (A_5,B_2) then forecast Y_A(September-1-1996) = ?"

Like before, after using the rule 26 listed in Table 6, a forecasted value (i.e. 28.0) is got for the forecasted value for the above pattern.

By the method, then we compare the above pattern with all λ-order fuzzy forecast rules and get a forecasted value of the matched one. The complete forecasted results based on the two-factor second-order fuzzy forecast rules in Table 6 are listed in Table 7.

3. Particle swarm optimization

The particle swarm optimization (PSO) is a novel multi-agent optimization system (MAOS) inspired by social behavior metaphor developed by Kennedy and Eberhart (Eberhart & Shi, 1998; Kennedy, Eberhart, & Shi, 2001). It is unlike an evolutionary algorithm, however, in that each potential solution is also assigned a randomized velocity, and the potential solutions, called particles, are then "flown" through the multi-dimensional problem space. The PSO method was generally found to perform better than other algorithms (for example, genetic algorithm, memetic algorithm, antcolony systems, and shuffled frog leaping) in terms of success rate and solution quality (Elbeltagi, Hegazy, & Grierson, 2005). In PSO, instead of using more traditional genetic operators, each particle (individual) adjusts its "flying" according to its own flying experience and its companions' flying experience.

3.1. Standard particle swarm optimization

In the standard particle swarm optimization (Shi & Eberhart, 2001), each particle i has a position represented by a position

Table 6
The complete two-factors second-order fuzzy forecast rules.

Rule No.	Matching part	Forecasting part
1	if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_7), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_6)	then forecast $Y(t) = 28.5$
2	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_6), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_5)	then forecast $Y(t) = 29.5$
3	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_5), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_4)	then forecast $Y(t) = 28.5$
4	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_4), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_4)	then forecast $Y(t) = 28.5$
5	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_4), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_7)	then forecast $Y(t) = 29.5$
6	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_7), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_5)	then forecast $Y(t) = 28.5$
7	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_5), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_2)	then forecast $Y(t) = 27.5$
8	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_2), and $(F_A(t-1), F_B(t-1))$ equals (A_5,B_5)	then forecast $Y(t) = 28.5$
9	if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_5), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_4)	then forecast $Y(t) = 28.5$
10	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_4), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_4)	then forecast $Y(t) = 28.5$
11	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_4), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_6)	then forecast $Y(t) = 29.5$
12	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_6), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_4)	then forecast $Y(t) = 27.5$
13	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_4), and $(F_A(t-1), F_B(t-1))$ equals (A_5,B_6)	then forecast $Y(t) = 26.5$
14	if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_6), and $(F_A(t-1), F_B(t-1))$ equals (A_4,B_5)	then forecast $Y(t) = 27.5$
15	if $(F_A(t-2), F_B(t-2))$ equals (A_4,B_5), and $(F_A(t-1), F_B(t-1))$ equals (A_5,B_6)	then forecast $Y(t) = 27.5$
16	if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_6), and $(F_A(t-1), F_B(t-1))$ equals (A_5,B_3)	then forecast $Y(t) = 29.5$
17	if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_3), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_2)	then forecast $Y(t) = 29.5$
18	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_2), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_2)	then forecast $Y(t) = 29.5$
19	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_2), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_4)	then forecast $Y(t) = 28.5$
20	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_4), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_3)	then forecast $Y(t) = 28.5$
21	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_3), and $(F_A(t-1), F_B(t-1))$ equals (A_6,B_3)	then forecast $Y(t) = 29.0$
22	if $(F_A(t-2), F_B(t-2))$ equals (A_6,B_3), and $(F_A(t-1), F_B(t-1))$ equals (A_7,B_3)	then forecast $Y(t) = 27.5$
23	if $(F_A(t-2), F_B(t-2))$ equals (A_7,B_3), and $(F_A(t-1), F_B(t-1))$ equals (A_5,B_1)	then forecast $Y(t) = 26.5$
24	if $(F_A(t-2), F_B(t-2))$ equals (A_5,B_1), and $(F_A(t-1), F_B(t-1))$ equals (A_4,B_2)	then forecast $Y(t) = 26.5$
25	if $(F_A(t-2), F_B(t-2))$ equals (A_4,B_2), and $(F_A(t-1), F_B(t-1))$ equals (A_4,B_3)	then forecast $Y(t) = 27.5$
26	if $(F_A(t-2), F_B(t-2))$ equals (A_4,B_3), and $(F_A(t-1), F_B(t-1))$ equals (A_5,B_2)	then forecast $Y(t) = 28.0$ (calculated by case 3 of **Step 6**)

Table 7
The complete forecasted results based on the two-factors second-order fuzzy forecast rules in Table 6.

Days	Actual daily average temperature (°C)	Fuzzy sets $F_A(t)$	Actual daily cloud density (%)	Fuzzy sets $F_B(t)$	Matched rule no.	Forecasted daily average temperature (°C)
August-1-1996	27.1	A_5	100	B_7	Not forecasted	
August-2-1996	28.9	A_6	78	B_6	Not forecasted	
August-3-1996	28.9	A_6	68	B_5	1	28.5
August-4-1996	29.3	A_7	44	B_4	2	29.5
August-5-1996	28.8	A_6	56	B_4	3	28.5
August-6-1996	28.7	A_6	89	B_7	4	28.5
August-7-1996	29.0	A_7	71	B_5	5	29.5
August-8-1996	28.2	A_6	28	B_2	6	28.5
August-9-1996	27.0	A_5	70	B_5	7	27.5
August-10-1996	28.3	A_6	44	B_4	8	28.5
August-11-1996	28.9	A_6	48	B_4	9	28.5
August-12-1996	28.1	A_6	76	B_6	10	28.5
August-13-1996	29.9	A_7	50	B_4	11	29.5
August-14-1996	27.6	A_5	84	B_6	12	27.5
August-15-1996	26.8	A_4	69	B_5	13	26.5
August-16-1996	27.6	A_5	78	B_6	14	27.5
August-17-1996	27.9	A_5	39	B_3	15	27.5
August-18-1996	29.0	A_7	20	B_2	16	29.5
August-19-1996	29.2	A_7	24	B_2	17	29.5
August-20-1996	29.8	A_7	25	B_2	18	29.5
August-21-1996	29.6	A_7	19	B_2	18	29.5
August-22-1996	29.3	A_7	46	B_4	18	29.5
August-23-1996	28.0	A_6	41	B_3	19	28.5
August-24-1996	28.3	A_6	34	B_3	20	28.5
August-25-1996	28.6	A_6	29	B_3	21	29.0
August-26-1996	28.7	A_6	31	B_3	21	29.0
August-27-1996	29.0	A_7	41	B_3	21	29.0
August-28-1996	27.7	A_5	14	B_1	22	27.5
August-29-1996	26.2	A_4	28	B_2	23	26.5
August-30-1996	26.0	A_4	33	B_3	24	26.5
August-31-1996	27.7	A_5	26	B_2	25	27.5
September-1-1996	N/A	N/A	N/A	N/A	26	28.0

(Suppose we dont know historical data of the daily average temperature and the daily cloud density in September-1-1996).

vector P_i. A swarm of particles moves through a multi-dimensional problem space and each particle i with the velocity represented by a vector V_i. Each particle keeps track of its own best position in each iteration (or time cycle), which is associated with the best experience it has achieved and denotes P_{best}. The best position among all the particles obtained in the population denotes P_{Gbest}. For each iteration, the position P_{best} of its own best and the position P_{Gbest} of the best particle of swarm are calculated as the best fitness of all particles. Accordingly, each particle changing the velocity this way enables the particle P_i to search around its individual best position P_{best} and the global best position P_{Gbest} as follows:

$$V_i = \omega \times V_i + c_1 \times rand_1() \times (P_{best} - P_i) + c_2 \times rand_2() \times (P_{Gbest} - P_i), \quad (5)$$

where c_1 and c_2 are positive acceleration constants (usually $c_1 = c_2 = 2$), $rand_1()$ and $rand_2()$ are uniformly distributed random numbers in the range [0,1], and ω is an inertia weight employed as an improvement proposed by Shi & Eberhart (1998). Then the particle flies toward a new position according to Eq. (6).

$$P_i = P_i + V_i. \quad (6)$$

The whole running procedure of the standard PSO is described in Algorithm 1.

Algorithm 1 (*Standard PSO algorithm*).

1. initialize all particles' positions P_i and velocities V_i, for $1 \leqslant i \leqslant$ NumberOfParticles.
2. **while** the stop condition (the optimal solution is found or the maximal moving steps are reached) is not satisfied **do**
3. **for** particle i, ($1 \leqslant i \leqslant$ NumberOfParticles) **do**
4. calculate the fitness value of particle i.
5. update the personal best position of particle i according to the fitness value.
6. **end for**
7. update the global best position of all particles according to the fitness value.
8. **for** particle i, ($1 \leqslant i \leqslant$ NumberOfParticles) **do**
9. move particle i to another position according to Eqs. (5) and (6).
10. **end for**
11. **end while**

3.2. Turbulent particle swarm optimization

In the standard PSO, it has been shown that the trajectories of the particles oscillate in different sinusoidal waves and converge quickly, sometimes prematurely (Liu & Abraham, 2005). Such situations could occur even in the early stages of the search. In fact, this does not even guarantee that the algorithm has converged to a local minimum and it merely means that all the particles have converged to the best position discovered so far by the swarm. During each iteration, the particle is attracted toward the location of the best fitness achieved so far by the particle itself and by the location of the best fitness achieved so far across the whole swarm. In order to guide the particles effectively in the search space, the maximum moving distance during an iteration and its moving range must be clamped in between the maximum velocity. The method to drive those lazy particles (i.e. the particles' velocity is small than threshold) and so that they can explore a better solution, named turbulent particle swarm optimization (TPSO) which is shown as follows:

$$V_i = \begin{cases} V_{\max} & \text{if} \quad V_i > V_{\max}; \\ -V_{\max} & \text{if} \quad V_i < -V_{\max}; \\ -V_{\max} + 2 \times V_{\max} \times rand() & \text{if} \quad |V_i| < V_s; \\ V_i & \text{otherwise}, \end{cases} \quad (7)$$

where the term V_i is limited to the range $\pm V_{max}$, $rand()$ is a uniformly distributed random number in the range [0,1], V_s is the minimum velocity threshold, a threshold parameter to limit the minimum of the particles' velocity. If V_s is large, it will shorten the oscillation period, and it provides a great probability for the particles to across local optimal using the same number of iterations. But a large V_s compels particles in the quick "flying" state, which leads them not to search the local optimal and forcing them not to refine the search. In other words, a large V_s facilitates a global search, and a small V_s facilitates a local search.

The whole running procedure of the TPSO is described in Algorithm 2.

Algorithm 2 (*TPSO algorithm*).

1. initialize all particles' positions P_i and velocities V_i, for $1 \leqslant i \leqslant$ NumberOfParticles.
2. **while** the stop condition (the optimal solution is found or the maximal moving steps are reached) is not satisfied **do**
3. **for** particle i, ($1 \leqslant i \leqslant$ NumberOfParticles) **do**
4. calculate the fitness value of particle i.
5. update the personal best position of particle i according to the fitness value.
6. **end for**
7. update the global best position of all particles according to the fitness value.
8. **for** particle i, ($1 \leqslant i \leqslant$ NumberOfParticles) **do**
9. move particle $i$to another position according to Eqs. (5)–(7).
10. **end for**
11. **end while**

4. The new proposed forecast model

A new forecast model "modified turbulent particle swarm optimization", named MTPSO, consisting of the fuzzy time series and the TPSO, is proposed in this paper. It can deal with the time-invariant model for the temperature prediction and forecasting TAIFEX. We describe the overall procedure, and then provide details of each step of the forecasting process by means of an example. The detailed descriptions of the MTPSO model for the temperature prediction are given in the following.

In the MTPSO model, for the training phase, we use the modified turbulent particle swarm optimization to train all two-factorλth-other fuzzy forecast rules under all training data. Once all fuzzy forecast rules have been well trained, for the testing phase, we can use the MTPSO model to forecast the new testing data. The detailed descriptions of MTPSO model are given as follows.

Let the number of the intervals of the main-factor be x, the lower bound and the upper bound of the universe of discourse on historical data $Y_A(t)$ of the main-factor be b_0 and b_x, respectively. Let the number of the intervals of the second-factor be y, the lower bound and the upper bound of the universe of discourse on historical data $Y_B(t)$ of the second-factor be d_0 and d_y, respectively. A particle is a vector consisting of $x + y - 2$ elements (i.e. $b_1, b_2, \ldots, b_{j-1}, b_j, \ldots, b_{x-2}, b_{x-1}$, and $d_1, d_2, \ldots, d_{k-1}, d_k, \ldots, d_{y-2}, d_{y-1}$ where $1 < j \leqslant x-1$, $b_{j-1} \leqslant b_j$ and $1 < k \leqslant y-1$, $d_{k-1} \leqslant d_k$); based on these $x + y - 2$ elements, define the $x + y$ intervals as $u_1 = [b_0, b_1), u_2 = [b_1, b_2), \ldots, u_j = [b_{j-1}, b_j), \ldots, u_x = [b_{x-1}, b_x]$, and $v_1 = [d_0, d_1), v_2 = [d_1, d_2), \ldots, v_k = [d_{k-1}, d_k), \ldots, v_y = [d_{y-1}, d_y]$, respectively. If a particle moves to another position, the elements of the corresponding new vector need to be sorted to ensure that each element b_j ($1 < j \leqslant x-1$) and d_k ($1 < k \leqslant y-1$) arrange in an ascending order, respectively.

The MTPSO model exploits the intervals denoted by each particle to create an independent group of fuzzy forecast rules to forecast all main-factor historical training data and get the forecasted accuracy for each particle. In order to compare the performance of the proposed method with the existing methods, the average forecasting error rate (AFER) value shown in Eq. (8) is used to represent the forecasted accuracy of a particle for the training phase for the temperature prediction, and the mean square error (MSE) value shown in Eq. (9) is used to represent the forecasted accuracy of a particle for the training phase for forecasting TAIFEX, respec-

Table 8
The initial position of particles 1–5.

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	d_1	d_2	d_3	d_4	d_5	d_6	AFER
Particle 1	24.00	25.00	26.00	27.00	28.00	29.00	30.00	31.00	14.29	28.57	42.86	57.14	71.43	85.71	1.06
Particle 2	23.96	25.06	25.98	27.44	28.41	29.16	30.02	30.80	13.67	27.37	44.79	60.47	69.39	80.36	0.94
Particle 3	23.89	24.65	25.91	26.88	28.44	29.19	30.02	30.93	14.11	30.55	43.16	62.78	70.25	88.55	0.91
Particle 4	24.02	25.19	26.56	27.08	28.22	29.17	30.14	31.25	11.48	24.06	45.24	52.17	66.78	83.55	0.95
Particle 5	24.32	25.01	25.92	27.06	28.25	29.30	29.86	31.23	8.97	30.57	41.87	52.52	68.26	82.19	0.97

Table 9
The initial velocities of particles 1–5.

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	d_1	d_2	d_3	d_4	d_5	d_6
Particle 1	−1.67	3.81	−0.20	0.61	1.16	1.62	1.17	1.85	1.02	21.40	1.52	10.59	46.67	32.21
Particle 2	−1.82	0.88	−3.70	−2.46	3.03	1.68	−4.86	0.62	−4.54	40.50	−21.78	−43.50	−2.34	48.37
Particle 3	4.22	0.61	1.52	2.73	−3.94	−4.99	0.42	−4.93	−4.87	−30.43	28.71	11.86	−48.45	39.09
Particle 4	2.62	4.07	2.59	−1.19	−1.69	0.04	0.65	2.67	27.99	−1.59	30.22	−2.90	−29.72	7.96
Particle 5	1.67	1.77	4.43	2.70	2.37	3.66	4.91	0.04	12.91	29.26	−5.14	2.44	−32.85	−36.93

Table 10
The second positions of particles 1–5.

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	d_1	d_2	d_3	d_4	d_5	d_6	AFER
Particle 1	23.47	25.92	26.07	27.16	28.44	29.53	30.35	31.54	14.56	35.41	43.38	61.50	85.18	95.97	0.85
Particle 2	23.35	24.80	24.95	26.19	28.56	29.34	29.69	31.10	12.71	36.76	42.42	49.53	69.47	100.00	1.45
Particle 3	24.83	25.15	26.36	27.26	27.69	27.70	29.45	30.14	12.65	21.42	51.77	55.71	66.33	100.00	1.32
Particle 4	24.56	25.43	26.15	26.36	28.11	29.22	30.11	31.46	24.65	35.37	50.52	64.16	70.58	95.03	1.16
Particle 5	24.31	25.11	27.23	27.66	29.18	30.27	30.89	31.52	18.89	39.32	41.84	60.75	65.33	78.60	1.30

Table 11
The second velocities of particles 1–5.

	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_8	d_1	d_2	d_3	d_4	d_5	d_6
Particle 1	−0.53	1.07	−0.08	0.16	0.44	0.53	0.35	0.54	0.27	6.84	0.52	4.36	13.75	10.26
Particle 2	−0.62	−0.11	−1.18	−1.24	0.94	0.53	−1.47	0.30	−0.97	15.05	−8.03	−10.94	0.08	22.00
Particle 3	1.27	0.18	0.46	0.82	−1.18	−1.50	0.13	−1.48	−1.46	−9.13	8.61	3.56	−14.53	11.73
Particle 4	0.54	0.23	−0.41	−0.72	−0.10	0.05	−0.03	0.21	13.16	11.31	5.28	18.41	−2.62	11.48
Particle 5	−0.01	0.10	1.31	0.61	0.93	0.96	1.65	−0.34	9.92	8.76	−0.02	12.81	−7.51	−3.59

tively. The lower the AFER (or MSE) value is, the better the forecasted accuracy is.

$$AFER = \frac{\sum_{i=1}^{n} \frac{|\text{forecasted value of day } i - \text{actual value of day } i|}{\text{actual value of day } i}}{n} \times 100\%. \quad (8)$$

$$MSE = \frac{\sum_{i=1}^{n} (\text{forecasted value of day } i - \text{actual value of day } i)^2}{n}. \quad (9)$$

For the training phase, the MTPSO algorithm moves all particles to another position according to Eqs. (5)–(7) and the intervals of the main-factor and the second-factor of all particles are sorted in an ascending order, respectively. Then repeat the steps mentioned above to evaluate the forecasted accuracy of all particles until the pre-defined stop condition (the optimal solution is found or the maximal moving steps are reached) is satisfied. If the stop condition is satisfied, then all two-factor λth-other fuzzy forecast rules trained by the best one of all personal best positions of all particles are chosen to be the final result. For the testing phase, the MTPSO algorithm uses all well trained two-factor λth-other fuzzy forecast rules to forecast the new testing data. The testing phase of the MTPSO algorithm is to perform a two-factor λth-other fuzzy forecast rules table searching problem. The detailed procedure of the MTPSO algorithm for the training phase and the testing phase are described in Algorithms 3 and 4, respectively.

Algorithm 3 (*The MTPSO algorithm for training phase*).

1. initialize all particles' positions P_i and velocities V_i, for $1 \leqslant i \leqslant$ NumberOfParticles.
2. **while** the stop condition (the optimal solution is found or the maximal moving steps are reached) is not satisfied **do**
3. **for** particle i, $(1 \leqslant i \leqslant$ NumberOfParticles$)$ **do**
4. fuzzify all historical training data including the main-factor and the second-factor, according to all intervals defined by the current position of particle i.
5. construct all two-factor λth-order fuzzy relationship groups according to all fuzzified historical training data.
6. create all fuzzy forecast rules based on all two-factor λth-order fuzzy relationships.
7. forecast all historical training data according to all two-factor λth-order fuzzy forecast rules.
8. calculate the fitness value of particle i based on Eq. (8) or Eq. (9).
9. update the personal best position of particle i according to the fitness value mentioned above.
10. **end for**
11. update the global position of all particles according to the fitness value mentioned above.
12. **for** particle i, $(1 \leqslant i \leqslant$ NumberOfParticles$)$ **do**

13. move particle i to another position according to Eqs. (5)–(7).
14. restrict the intervals of the main-factor and the second-factor to the lower bound and the upper bound of the universe of discourse within particle i, respectively. That is, any value updated from Eqs. (5)–(7) is smaller (larger) than the lower (upper) bound is set to the lower (upper) bound.
15. sort the intervals of the main-factor and the second-factor within particle $i$in an ascending order, respectively.
16. **end for**
17. **end while**

Algorithm 4 (*The MTPSO algorithm for testing phase*).

1. **if** the corresponding matching pattern with respect to the new testing data matches a trained fuzzy forecast rule.
2. **then** forecast the new testing data based on the forecasting part of the matched fuzzy forecast rule.

Table 12
A comparison of the average forecasting error rate of the proposed method with those of the existing methods in June 1996 in the training phase.

	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
The proposed method								
Average	0.51%	0.36%	0.36%	0.33%	0.32%	0.33%	0.28%	0.30%
Minimum	0.45%	0.36%	0.34%	0.32%	0.31%	0.31%	0.28%	0.29%
Standard deviation	0.04%	0.01%	0.03%	0.01%	0.02%	0.03%	0.01%	0.02%
Lee et al.'s method (Lee et al., 2008)								
Annealing constant	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
$\alpha = 0.25$	0.79%	0.44%	0.42%	0.42%	0.42%	0.44%	0.40%	0.40%
$\alpha = 0.5$	0.84%	0.50%	0.45%	0.42%	0.38%	0.43%	0.39%	0.46%
$\alpha = 0.9$	0.79%	0.46%	0.42%	0.44%	0.42%	0.41%	0.46%	0.39%
Lee et al.'s method (Lee et al., 2007)								
Window basis	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
	1.24%	0.74%	0.64%	0.72%	0.65%	0.66%	0.64%	0.65%
Chen et al.'s method (Chen & Hwang, 2000)								
Window basis		$w = 2$	$w = 3$	$w = 4$	$w = 5$	$w = 6$	$w = 7$	$w = 8$
		2.88%	3.16%	3.24%	3.33%	3.39%	3.53%	3.67%

Table 13
A comparison of the average forecasting error rate of the proposed method with those of the existing methods in July 1996 in the training phase.

	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
The proposed method								
Average	0.50%	0.36%	0.35%	0.36%	0.34%	0.34%	0.36%	0.34%
Minimum	0.41%	0.34%	0.33%	0.33%	0.32%	0.32%	0.34%	0.33%
Standard deviation	0.04%	0.03%	0.02%	0.02%	0.02%	0.02%	0.03%	0.01%
Lee et al.'s method (Lee et al., 2008)								
Annealing constant	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
$\alpha = 0.25$	0.66%	0.45%	0.42%	0.41%	0.41%	0.40%	0.41%	0.40%
$\alpha = 0.5$	0.66%	0.50%	0.47%	0.44%	0.40%	0.38%	0.44%	0.42%
$\alpha = 0.9$	0.62%	0.46%	0.45%	0.44%	0.44%	0.41%	0.40%	0.40%
Lee et al.'s method (Lee et al., 2007)								
Window basis	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
	1.23%	0.78%	0.73%	0.83%	0.70%	0.71%	0.68%	0.69%
Chen et al.'s method (Chen & Hwang, 2000)								
Window basis		$w = 2$	$w = 3$	$w = 4$	$w = 5$	$w = 6$	$w = 7$	$w = 8$
		3.04%	3.76%	4.08%	4.17%	4.35%	4.38%	4.56%

Table 14
A comparison of the average forecasting error rate of the proposed method with those of the existing methods in August 1996 in the training phase.

	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
The proposed method								
Average	0.48%	0.32%	0.35%	0.34%	0.33%	0.34%	0.35%	0.36%
Minimum	0.41%	0.31%	0.34%	0.33%	0.33%	0.33%	0.34%	0.35%
Standard deviation	0.05%	0.03%	0.00%	0.00%	0.00%	0.01%	0.01%	0.00%
Lee et al.'s method (Lee et al., 2008)								
Annealing constant	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
$\alpha = 0.25$	0.64%	0.43%	0.47%	0.40%	0.41%	0.38%	0.40%	0.45%
$\alpha = 0.5$	0.69%	0.40%	0.38%	0.37%	0.37%	0.39%	0.42%	0.45%
$\alpha = 0.9$	0.66%	0.40%	0.40%	0.40%	0.36%	0.41%	0.39%	0.44%
Lee et al.'s method (Lee et al., 2007)								
Window basis	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
	1.09%	0.92%	0.88%	1.07%	0.75%	0.76%	0.75%	0.73%
Chen et al.'s method (Chen & Hwang, 2000)								
Window basis		$w = 2$	$w = 3$	$w = 4$	$w = 5$	$w = 6$	$w = 7$	$w = 8$
		2.75%	2.77%	3.30%	3.40%	3.18%	3.15%	3.19%

3. **else** forecast the new testing data based on Eq. (4) proposed in Section 2.
4. **end if**

An example for illustrating the MTPSO model during the training phase is given in the following. In this example, the MTPSO model includes the TPSO algorithm to train all two-factor second-order fuzzy forecast rules under all historical training data (i.e. $Y_A(t)$ and $Y_B(t)$ in Table 3, August-1-1996 $\leqslant t \leqslant$ August-31-1996). The symbols b_j for $1 \leqslant j \leqslant 8$ denotes the interval bound of the universe of discourse on $Y_A(t)$, and d_k for $1 \leqslant k \leqslant 6$ denotes the interval bound of the universe of discourse on $Y_B(t)$. For the symbols in Eqs. (5)–(7), a position vector P_i consists of two parts b_j ($1 \leqslant j \leqslant 8$) and d_k ($1 \leqslant k \leqslant 6$), where the values of the former and the latter are limited to [23.0, 32.0] and [0.0, 100.0] respectively; the values of the velocity vector V_i corresponding to the former are limited to $[-5,5]$ and those corresponding to the latter are limited to $[-50,50]$; the value of the velocity threshold V_s corresponding to the former is set to 0.001 and that corresponding to the latter is set to 0.005, both c_1 and c_2 are set to 2, and ω is set to 0.3. Let the number of particles be 5. The randomized initial positions and the initial velocities of all particles are listed in Tables 8 and 9, respectively.

In Table 8, each particle defines an independent group of sixteen intervals which are $u_1 = [b_0, b_1), u_2 = [b_1, b_2), \ldots, u_9 = [b_8, b_9]$ and $v_1 = [d_0, d_1), v_2 = [d_1, d_2), \ldots, v_7 = [d_6, d_7]$, respectively. For example, the intervals of the initial position of particle 1 listed in Table 8 are then $u_1 = [23.0, 24.0), u_2 = [24.0, 25.0), \ldots, u_8 = [30.0, 31.0), u_9 = [31.0, 32.0]$ and $v_1 = [0.0, 14.29)$, $v_2 = [14.29, 28.57), \ldots, v_6 = 71.43, 85.71), v_7 = [85.71, 100.0]$, respectively.

For simplicity, assume that 9 intervals for the main-factor and 7 intervals for the second-factor created by particle 1 are identical to the one used in the forecasting example mentioned in Section 2. So, we follow the whole forecasting procedure described in Section 2 with respect to the corresponding steps in Algorithm 3, fuzzify all historical data listed in Table 3, construct all two-factor second-order fuzzy relationship groups listed in Table 4, get an independent group of the trained two-factor second-order fuzzy forecast rules listed in Table 5, and an individual group of the forecasted results listed in Tables 6 and 7, respectively. And the AFER value for particle 1 in Table 8 is calculated in Eq. (10) based on Eq. (8) as follows,

Table 16
Historical data of the TAIFEX index and the TAIEX index from 8/3/1998 to 9/30/1998 Huarng (2001b).

Date	TAIFEX index	TAIEX index
8/3/1998	7552	7599
8/4/1998	7560	7593
8/5/1998	7487	7500
8/6/1998	7462	7472
8/7/1998	7515	7530
8/10/1998	7365	7372
8/11/1998	7360	7384
8/12/1998	7330	7352
8/13/1998	7291	7363
8/14/1998	7320	7348
8/15/1998	7300	7372
8/17/1998	7219	7274
8/18/1998	7220	7182
8/19/1998	7285	7293
8/20/1998	7274	7271
8/21/1998	7225	7213
8/24/1998	6955	6958
8/25/1998	6949	6908
8/26/1998	6790	6814
8/27/1998	6835	6813
8/28/1998	6695	6724
8/29/1998	6728	6736
8/31/1998	6566	6550
9/1/1998	6409	6335
9/2/1998	6430	6472
9/3/1998	6200	6251
9/4/1998	6403.2	6463
9/5/1998	6697.5	6756
9/7/1998	6722.3	6801
9/8/1998	6859.4	6942
9/9/1998	6769.6	6895
9/10/1998	6709.75	6804
9/11/1998	6726.5	6842
9/14/1998	6774.55	6860
9/15/1998	6762	6858
9/16/1998	6952.75	6973
9/17/1998	6906	7001
9/18/1998	6842	6962
9/19/1998	7039	7150
9/21/1998	6861	7029
9/22/1998	6926	7034
9/23/1998	6852	6962
9/24/1998	6890	6980
9/25/1998	6871	6980
9/28/1998	6840	6911
9/29/1998	6806	6885
9/30/1998	6787	6834

Table 15
A comparison of the average forecasting error rate of the proposed method with those of the existing methods in September 1996 in the training phase.

	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
The proposed method								
Average	0.56%	0.55%	0.57%	0.54%	0.51%	0.52%	0.53%	0.45%
Minimum	0.54%	0.54%	0.56%	0.54%	0.50%	0.51%	0.52%	0.41%
Standard deviation	0.02%	0.00%	0.00%	0.00%	0.01%	0.01%	0.01%	0.02%
Lee et al.'s method (Lee et al., 2008)								
Annealing constant	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
$\alpha = 0.25$	0.69%	0.58%	0.59%	0.57%	0.56%	0.57%	0.58%	0.47%
$\alpha = 0.5$	0.66%	0.62%	0.59%	0.59%	0.56%	0.54%	0.56%	0.53%
$\alpha = 0.9$	0.62%	0.59%	0.61%	0.57%	0.54%	0.59%	0.57%	0.50%
Lee et al.'s method (Lee et al., 2007)								
Window basis	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
	1.28%	0.91%	0.86%	1.03%	0.87%	0.97%	0.84%	0.82%
Chen et al.'s method (Chen & Hwang, 2000)								
Window basis		$w = 2$	$w = 3$	$w = 4$	$w = 5$	$w = 6$	$w = 7$	$w = 8$
		3.29%	3.10%	3.19%	3.22%	3.39%	3.38%	3.29%

Table 17
A comparison of the mean square error of the proposed method with that of Lee et al.'s method for the training phase.

	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
The proposed method								
Average	875.84	177.43	167.91	126.97	116.97	118.96	100.15	108.87
Minimum	530.80	153.48	140.97	112.41	105.40	114.81	92.17	94.54
Standard deviation	215.19	22.00	12.80	15.42	8.54	2.15	6.62	10.43
Lee et al.'s method (Lee et al., 2008)								
Annealing constant	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
$\alpha = 0.25$	849.03	192.57	282.71	202.87	226.67	309.18	316.40	151.34
$\alpha = 0.5$	1205.32	213.47	280.82	206.48	217.84	176.51	244.51	148.63
$\alpha = 0.9$	799.19	193.88	208.79	142.26	143.31	147.14	105.02	124.45
Lee et al.'s method (Lee et al., 2007)								
Window basis	1st-order	2nd-order	3rd-order	4th-order	5th-order	6th-order	7th-order	8th-order
	1779.30	667.57	543.74	530.63	722.09	305.75	249.61	251.65

$$
\begin{aligned}
AFER &= \frac{\sum_{i=1}^{n}\frac{|\text{forecasted value of day } i - \text{actual value of day } i|}{\text{actual value of day } i}}{n}\times 100\% \\
&= \frac{\sum_{i=3}^{31}\frac{|\text{forecasted value of day } i - \text{actual value of day } i|}{\text{actual value of day } i}}{29}\times 100\% \\
&= \frac{\frac{|28.5-28.9|}{28.9}+\frac{|29.5-29.3|}{29.3}+\frac{|28.5-28.8|}{28.8}+\cdots+\frac{|26.5-26.0|}{26.0}+\frac{|27.5-27.7|}{27.7}}{29} \\
&\quad \times 100\% \\
&= 1.06\%,
\end{aligned} \tag{10}
$$

where n denotes the number of the forecasted data (i.e. 29), and $3 \leqslant i \leqslant 31$ (i.e. from August-3-1996 to August-31-1996).

After all particles have got their own AFER values, every particle updates its own personal best position so far according to the AFER value. Note that the initial personal best positions are set as the initial positions of all particles. The personal best positions of all particles so far are listed in Table 8. In Table 8, since the AFTER value of particle 3 is the least among all five particles so far, the global best position (i.e. P_{Gbest} in Eq. (5)) is then set to particle 3. Then the MTPSO algorithm moves all particles to the second positions according to Eqs. (5)–(7).

$$
\begin{aligned}
b_1 &= 0.3\times(-1.67)+2\times rand_1()\times(24.00-24.00)+2\times rand_2()\times(23.89-24.00) = -0.53;\\
b_2 &= 0.3\times(3.81)+2\times rand_1()\times(25.00-25.00)+2\times rand_2()\times(24.65-25.00) = 1.07;\\
b_3 &= 0.3\times(-0.20)+2\times rand_1()\times(26.00-26.00)+2\times rand_2()\times(25.91-26.00) = -0.08;\\
b_4 &= 0.3\times(0.61)+2\times rand_1()\times(27.00-27.00)+2\times rand_2()\times(26.88-27.00) = 0.16;\\
b_5 &= 0.3\times(1.16)+2\times rand_1()\times(28.00-28.00)+2\times rand_2()\times(28.44-28.00) = 0.44;\\
b_6 &= 0.3\times(1.62)+2\times rand_1()\times(29.00-29.00)+2\times rand_2()\times(29.19-29.00) = 0.53;\\
b_7 &= 0.3\times(1.17)+2\times rand_1()\times(30.00-30.00)+2\times rand_2()\times(30.02-30.00) = 0.35;\\
b_8 &= 0.3\times(1.85)+2\times rand_1()\times(31.00-31.00)+2\times rand_2()\times(30.93-31.00) = 0.54;\\
d_1 &= 0.3\times(1.02)+2\times rand_1()\times(14.29-14.29)+2\times rand_2()\times(14.11-14.29) = 0.27;\\
d_2 &= 0.3\times(21.40)+2\times rand_1()\times(28.57-28.57)+2\times rand_2()\times(30.55-28.57) = 6.84;\\
d_3 &= 0.3\times(1.52)+2\times rand_1()\times(42.86-42.86)+2\times rand_2()\times(43.16-42.86) = 0.52;\\
d_4 &= 0.3\times(10.59)+2\times rand_1()\times(57.14-57.14)+2\times rand_2()\times(62.78-57.14) = 4.36;\\
d_5 &= 0.3\times(46.67)+2\times rand_1()\times(71.43-71.43)+2\times rand_2()\times(70.25-71.43) = 13.75;\\
d_6 &= 0.3\times(32.21)+2\times rand_1()\times(85.71-85.71)+2\times rand_2()\times(88.55-85.71) = 10.26.
\end{aligned} \tag{11}
$$

The velocity of particle 1 is calculated in Eq. (11) based on Eq. (5). In Eq. (11) the elements will be restricted by Eq. (7) (i.e. velocity vector V_i that $b_j \in V_i$ $(1 \leqslant j \leqslant 8)$ is limited to $[-5,5]$ and $d_k \in V_i$ $(1 \leqslant k \leqslant 6)$ is limited to $[-50, 50]$). In other words, the velocity vector is = {−0.53, 1.07, −0.08, 0.16, 0.44, 0.53, 0.35, 0.54, 0.27, 6.84, 0.52, 4.36, 13.75, 10.26}. Then the position vector of particle 1 = {23.47, 26.07, 25.92, 27.16, 28.44, 29.53, 30.35, 31.54, 14.56, 35.41, 43.38, 61.50, 85.18, 95.97} based on Eq. (6). Last, restrict the main-factor (i.e. $Y_A(t)$) and the second-factor (i.e. $Y_B(t)$) in the lower bound and upper bound of the universe of discourse within particle 1, respectively, and intervals of corresponding new position vector of the main-factor and the second-factor within particle *i*need to be sorted to ensure that each interval b_j $(1 \leq j \leq 8)$ and d_k $(1 \leq k \leq 6)$ arrange in an ascending order, respectively. By this procedure, we can get the other particles' position and velocity. Now the MTPSO model moves all particles to the second positions according to Eqs. (5)–(7). The second positions and the corresponding new AFER values of all particles are listed in Table 10, and the second velocities of all particles are listed in Table 11.

By comparing the AFER values listed in Table 8 with those listed in Table 10, it is obvious that particle 1 reaches a better position than its own personal best position so far. Also the new global best

Table 18
A comparison of the mean square error of the proposed method with those of other existing methods in the training phase.

Date	Actual TAIFEX index	Chen's method Chen (1996)	Huarng's method (Two-variable heuristic)Huarng (2001a)	Huarng's method (Three-variable heuristic) Huarng (2001b)	Lee et al.'s method Lee et al. (2006)	Lee et al.'s method Lee et al. (2007)	Lee et al.'s method Lee et al. (2008)	The proposed method (Seventh-order; MTPSO)
8/3/1998	7552							
8/4/1998	7560	7450	7450	7450				
8/5/1998	7487	7450	7450	7450				
8/6/1998	7462	7500	7450	7500	7450			
8/7/1998	7515	7500	7500	7500	7550			
8/10/1998	7365	7450	7450	7450	7350			
8/11/1998	7360	7300	7350	7300	7350			
8/12/1998	7330	7300	7300	7300	7350	7348	7329	7325.28
8/13/1998	7291	7300	7350	7300	7250	7301.5	7289.5	7287.48
8/14/1998	7320	7183.33	7100	7188.33	7350	7311.5	7329	7325.28
8/15/1998	7300	7300	7350	7300	7350	7301.5	7289.5	7287.48
8/17/1998	7219	7300	7300	7300	7250	7226.5	7215	7221.26
8/18/1998	7220	7183.33	7100	7100	7250	7226.5	7215	7221.26
8/19/1998	7285	7183.33	7300	7300	7250	7301.5	7289.5	7287.48
8/20/1998	7274	7183.33	7100	7188.33	7250	7256.5	7289.5	7287.48
8/21/1998	7225	7183.33	7100	7100	7250	7226.5	7215	7221.26
8/24/1998	6955	7183.33	7100	7100	6950	6952	6949.5	6952.02
8/25/1998	6949	6850	6850	6850	6950	6952	6949.5	6952.02
8/26/1998	6790	6850	6850	6850	6750	6783.5	6796	6781.01
8/27/1998	6835	6775	6650	6775	6850	6852	6848	6842.05
8/28/1998	6695	6850	6750	6750	6650	6713	6698.5	6696.17
8/29/1998	6728	6750	6750	6750	6750	6713	6726	6726.50
8/31/1998	6566	6775	6650	6650	6550	6561	6569.5	6580.45
9/1/1998	6409	6450	6450	6450	6450	6406	6417	6409.24
9/2/1998	6430	6450	6550	6550	6450	6406	6417	6409.24
9/3/1998	6200	6450	6350	6350	6250	6198.5	6205	6213.94
9/4/1998	6403.2	6450	6450	6450	6450	6406	6417	6409.24
9/5/1998	6697.5	6450	6550	6550	6650	6703	6698.5	6696.17
9/7/1998	6722.3	6750	6750	6750	6750	6713	6726	6726.50
9/8/1998	6859.4	6775	6850	6850	6850	6852	6848	6864.96
9/9/1998	6769.6	6850	6750	6750	6750	6783.5	6763	6781.01
9/10/1998	6709.75	6775	6650	6650	6750	6713	6726	6696.17
9/11/1998	6726.5	6775	6850	6775	6750	6713	6726	6726.50
9/14/1998	6774.55	6775	6850	6775	6817	6783.5	6763	6781.01
9/15/1998	6762	6775	6650	6775	6817	6783.5	6763	6781.01
9/16/1998	6952.75	6775	6850	6850	6817	6953	6949.5	6952.02
9/17/1998	6906	6850	6950	6850	6950	6952	6904.5	6906.70
9/18/1998	6842	6850	6850	6850	6850	6852	6848	6842.05
9/19/1998	7039	6850	6950	6850	7050	7089	7064	7039.20
9/21/1998	6861	6850	6850	6850	6850	6852	6848	6864.96
9/22/1998	6926	6850	6950	6850	6950	6952	6904.5	6906.70
9/23/1998	6852	6850	6850	6850	6850	6852	6848	6842.05
9/24/1998	6890	6850	6950	6850	6850	6893	6904.5	6906.70
9/25/1998	6871	6850	6850	6850	6850	6852	6848	6864.96
9/28/1998	6840	6850	6750	6750	6850	6852	6848	6842.05
9/29/1998	6806	6850	6750	6850	6850	6792.5	6796	6781.01
9/30/1998	6787	6850	6750	6750	6750	6783.5	6796	6781.01
MSE		9668.94	7856.5	5437.58	1364.56	249.61	105.02	92.17

Table 19
A comparison of the mean square errors of the proposed method under two-factor seventh-order fuzzy relationships forecasting model to those of Lee et al.'s methods in the testing phase.[a]

	Actual TAIFEX index	Lee et al.'s method Lee et al. (2007)					Lee et al.'s method Lee et al. (2008)					The proposed method				
		R1	R2	R3	R4	R5	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5
9/10/1998	6709.75	6618.01	6611.68	6615.48	6614.86	6621.43	6928.23	6898.27	6918.74	6917.40	6940.32	6738.53	6726.08	6734.51	6733.95	6745.45
9/11/1998	6726.5	6683.42	6678.66	6677.80	6678.52	6677.48	6843.59	6847.00	6833.64	6852.23	6844.41	6764.25	6759.73	6756.24	6760.57	6757.89
9/14/1998	6774.55	6711.88	6714.55	6714.02	6696.02	6709.63	6799.85	6794.92	6810.33	6805.71	6801.85	6734.07	6736.45	6739.72	6718.22	6731.76
9/15/1998	6762	6735.55	6725.65	6732.89	6734.10	6732.02	6763.78	6770.44	6760.34	6762.37	6757.97	6726.55	6722.37	6723.91	6725.12	6722.54
9/16/1998	6952.75	6751.10	6751.99	6754.04	6753.79	6753.38	6791.52	6786.10	6789.71	6793.06	6787.25	6752.84	6752.16	6754.75	6755.38	6753.72
9/17/1998	6906	6759.98	6756.68	6757.13	6758.73	6756.02	6788.16	6786.01	6785.15	6784.40	6782.97	6765.37	6762.75	6762.95	6763.46	6761.54
9/18/1998	6842	6813.32	6802.42	6805.79	6805.88	6804.26	6964.29	6958.33	6958.09	6970.74	6970.06	6864.83	6853.45	6855.95	6859.65	6857.27
9/19/1998	7039	6836.26	6835.09	6838.97	6842.18	6842.04	6962.23	6928.86	6964.39	6977.22	6968.04	6890.64	6882.64	6893.47	6900.53	6898.97
9/21/1998	6861	6845.38	6850.54	6848.56	6853.19	6839.01	6867.86	6860.38	6871.17	6874.46	6854.84	6862.36	6868.10	6868.70	6873.82	6853.07
9/22/1998	6926	6904.72	6907.21	6910.99	6905.26	6897.33	7083.07	7107.55	7105.86	7126.05	7094.50	6951.21	6958.78	6961.68	6960.41	6951.95
9/23/1998	6852	6903.37	6908.97	6904.24	6907.43	6896.83	6857.20	6868.28	6862.71	6862.49	6858.96	6899.55	6909.34	6901.60	6906.14	6896.84
9/24/1998	6890	6918.69	6907.90	6915.33	6912.33	6919.27	6969.43	6932.60	6959.07	6944.36	6958.77	6921.17	6904.29	6916.62	6909.81	6919.94
9/25/1998	6871	6900.59	6898.23	6906.50	6905.18	6903.36	6832.85	6826.99	6833.52	6831.88	6832.86	6882.44	6878.28	6886.00	6885.13	6884.99
9/28/1998	6840	6908.27	6900.22	6903.14	6894.83	6895.95	6936.67	6900.98	6896.95	6843.24	6911.98	6910.55	6893.01	6892.40	6874.54	6894.10
9/29/1998	6806	6882.39	6883.24	6887.30	6881.62	6879.31	6855.42	6859.03	6863.76	6858.45	6860.31	6864.21	6866.85	6871.54	6865.87	6866.17
9/30/1998	6787	6868.88	6872.16	6868.67	6868.02	6878.34	6821.64	6826.67	6823.38	6825.64	6806.04	6858.44	6862.44	6859.12	6858.38	6865.06
MSE		8815.32	8815.32	9012.04	8813.86	8755.44	10519.78	10519.78	10216.53	10207.16	10601.61	6555.68	6555.68	6662.58	6385.86	6236.40
RMSE[b]		93.89	93.89	94.93	93.88	93.57	102.57	102.57	101.08	101.03	102.96	80.97	80.97	81.62	79.91	78.97
Average of RMSE				94.03					102.04					80.49		
Minimum of RMSE				93.57					101.03					78.97		
Standard deviation of RMSE				0.52					0.92					1.05		

[a] All three methods use the same number of intervals.
[b] The function of root mean squared error (RMSE) is defined by: $RMSE = \sqrt{\frac{\sum_{i=1}^{n}(\text{forecasted value of day}i - \text{actual value of day}i)^2}{n}}$.

position for all particles is created by particle 1 as its AFER is the least for all particles so far. The MTPSO model repeats the above steps until the stop condition is satisfied.

5. Experimental results

5.1. Experimental results for the training phase

The essential parameters of MTPSO model for the temperature prediction are set as follows. We simulated 10 runs in each order. Let the number of iterations be 1000, the number of particles be 30, the value of inertial weight (i.e. ω) be 0.3, the self confidence coefficient (i.e. c_1) and the social confidence coefficient (i.e. c_2) both be 2, the velocity of the main-factor (i.e., the daily average temperature) be limited to $[-5, 5]$, the minimum velocity threshold of the main-factor V_{s1} be 0.001, the velocity of the second-factor (i.e., the daily cloud density) be limited to $[-50, 50]$, the minimum velocity threshold of the second-factor V_{s2} be 0.005, the universe of discourse on Y_A of the main-factor on the fuzzy time series be $[23.0, 32.0]$ and be cut into 9 intervals, and the universe of discourse on Y_B of the second-factor on the fuzzy time series be $[0.0, 100.0]$ and be cut into 7 intervals, respectively.

A comparison of the forecasted accuracy (i.e. the AFER value) between the proposed method, Lee et al. (2007, 2008) method and Chen and Hwang (2000) method under different order fuzzy relationships is listed in Tables 12–15. We can see that the proposed method gets smaller forecasting error rate than the methods presented in Chen and Hwang (2000) and Lee et al. (2007, 2008). That is, the proposed method gets the higher forecasting accuracy rate than the methods presented in Chen and Hwang (2000) and Lee et al. (2007, 2008) for dealing with the temperature prediction from June 1996 to September 1996 in Taipei, Taiwan.

The essential parameters of MTPSO model for forecasting TAIFEX are set as follows. We simulated 10 runs in each order. Let the number of iterations be 1000, the number of particles be 30, the value of inertial weight (i.e. ω) be 0.3, the self confidence coefficient (i.e. c_1) and the social confidence coefficient (i.e. c_2) both be 2, the velocity of the main-factor (i.e., the TAIFEX) be limited to $[-750, 750]$, the minimum velocity threshold of the main-factor V_{s1} be 0.1, the velocity of the second-factor (i.e., the TAIEX) be limited to $[-750, 750]$, the minimum velocity threshold of the second-factor V_{s2} be 0.1, the universe of discourse on Y_A of the main-factor on the fuzzy time series be $[6100, 7700]$ and be cut into 16 intervals, and the universe of discourse on Y_B of the second-factor on the fuzzy time series be $[6100, 7700]$ and be cut into 16 intervals, respectively. Let $Y_A(t)$ and $Y_B(t)$ be two historical data of the TAIFEX index and the TAIEX index on day t ($8/3/1998 \leqslant t \leqslant 9/30/1998$) is listed in Table 16.

A comparison of the forecasted accuracy (i.e. the MSE value) between the proposed method and the existing methods under different order fuzzy relationships is listed in Table 17 (Lee et al., 2007, 2008) and 18 (Chen, 1996; Huarng, 2001a, 2001b; Lee et al., 2006, 2007, 2008). By Tables 17 and 18, the proposed method also gets the smaller forecasting error rate than the methods presented in Chen (1996), Huarng (2001a, 2001b), and Lee et al. (2006, 2007, 2008). In Table 17, we list the comparison to Lee et. al.'s methods Lee et al. (2007, 2008) with different orders up to 8. In Table 18, we list the comparison to all existing methods each with the best forecasting accuracy.

5.2. Experimental results for the testing phase

In this section, we used the result of training phase to forecasting TAIFEX index. Based on the historical data for the past days, we can forecast the new TAIFEX index for the next day only. For example, the historical data of TAIFEX index and TAIFEX index under days 8/3/1998–9/23/1998, are used to forecast the new TAIFEX index of day 9/24/1998 using MTPSO. And the historical data of TAIFEX index and TAIFEX index under days 8/3/1998–9/24/1998, are used to forecast the new TAIFEX index of day 9/25/1998. In the experiment, based on two-factor seventh-order fuzzy relationship, we ran the MTPSO algorithm for five times (i.e. R1, R2, R3, R4, R5 in Table 19) and forecasted the latest 16 (i.e. from 9/10/1998 to 9/30/1998 in Table 19) of total 47 (i.e. from 8/3/1998 to 9/30/1998 in Table 16). The results of ours are also compared to those of Lee et. al.'s methods Lee et al. (2007, 2008). To forecast the untrained pattern, the formula $\frac{1\times m_{ik}+2\times m_{i(k-1)}+\cdots+k\times m_{i1}}{1+2+\cdots+k}$ is used in (Lee et al., 2007) and the formula $m_{ik}+\frac{(m_{i(k-1)}-m_{ik})+(m_{i(k-2)}-m_{i(k-1)})+\cdots+(m_{i1}-m_{i2})}{k-1}$ is used in (Lee et al., 2008), respectively.

Experimental results of forecasted accuracy (i.e. the MSE value) in the testing phase are listed in Table 19. As shown in Table 19 for forecasting TAIFEX index from 9/10/1998 to 9/30/1998, the proposed method gets the smaller forecasting error rate than those of the methods presented in Lee et al. (2007, 2008).

6. Conclusions

In this paper, we proposed a modified turbulent particle swarm optimization (named MTPSO) forecast model based on two-factor high-order fuzzy relationships and particle swarm optimization. The proposed method uses the MTPSO technique to adjust the length of each interval in the universe of discourse for the temperature prediction and the TAIFEX forecasting to improve the forecasting accuracy rate. The experimental results for these two problems show that the MTPSO model is more precise than any existing methods that forecast data for both the training phase and the testing phase.

References

Chen, S. M. (1996). Forecasting enrollments based on fuzzy time series. *Fuzzy Sets and Systems, 81*, 311–319.

Cheng, C. H., Chen, T. L., Teoh, H. J., & Chiang, C. H. (2008). Fuzzy time-series based on adaptive expectation model for TAIEX forecasting. *Expert Systems with Applications, 34*, 1126–1132.

Chen, S. M., & Hwang, J. R. (2000). Temperature prediction using fuzzy time series. *IEEE Transactions on Systems, Man, and Cybernetics, Part B, 30*, 263–275.

Chu, H. H., Chen, T. L., Cheng, C. H., & Huang, C. C. (2009). Fuzzy dual-factor time-series for stock index forecasting. *Expert Systems with Applications, 36*, 165–171.

Eberhart, R. C., & Shi, Y. (1998). *Comparison between Genetic Algorithms and Particle Swarm Optimization* (Vol. 1447). Berlin, Heidelberg: Springer.

Elbeltagi, E., Hegazy, T., & Grierson, D. (2005). Comparison among five evolutionary-based optimization algorithms. *Advanced Engineering Informatics, 19*, 43–53.

Huarng, K. (2001a). Effective lengths of intervals to improve forecasting in fuzzy time series. *Fuzzy Sets and Systems, 123*, 387–394.

Huarng, K. (2001b). Heuristic models of fuzzy time series for forecasting. *Fuzzy Sets and Systems, 123*, 369–386.

Huarng, K., & Yu, H. K. (2005). A type 2 fuzzy time series model for stock index forecasting. *Physica A: Statistical Mechanics and its Applications, 353*, 445–462.

Kennedy, J., Eberhart, R. C., & Shi, Y. (2001). The particle swarm. In *Swarm intelligence* (pp. 287–325). San Francisco: Morgan Kaufmann.

Kuo, I. H., Horng, S. J., Kao, T. W., Lin, T. L., Lee, C. L., & Pan, Y. (2009). An improved method for forecasting enrollments based on fuzzy time series and particle swarm optimization. *Expert Systems with Applications, 36*, 6108–6117.

Lee, L. W., Wang, L. H., & Chen, S. M. (2007). Temperature prediction and TAIFEX forecasting based on fuzzy logical relationships and genetic algorithms. *Expert Systems with Applications, 33*, 539–550.

Lee, L. W., Wang, L. H., & Chen, S. M. (2008). Temperature prediction and TAIFEX forecasting based on high-order fuzzy logical relationships and genetic simulated annealing techniques. *Expert Systems with Applications, 34*, 328–336.

Lee, L. W., Wang, L. H., Chen, S. M., & Leu, Y. H. (2006). Handling forecasting problems based on two-factors high-order fuzzy time series. *IEEE Transactions on Fuzzy Systems, 14*, 468–477.

Li, Z., Chen, Z., & Li, J. (1988). A model of weather forecast by fuzzy grade statistics. *Fuzzy Sets and Systems, 26*, 275–281.

Liu, H., & Abraham, A. (2005). Fuzzy adaptive turbulent particle swarm optimization. In *Hybrid intelligent systems, 2005. HIS '05. Fifth international conference on, 2005* (p. 6).

Shi, Y., & Eberhart, R. C. (1998). A modified particle swarm optimizer. In *Evolutionary computation proceedings, 1998. IEEE world congress on computational intelligence, IEEE international conference on, 1998* (pp. 69–73).

Shi, Y., & Eberhart, R. C. (2001). Fuzzy adaptive particle swarm optimization. In *Evolutionary computation, 2001. Proceedings of the 2001 Congress on, 2001* (Vol. 1, pp. 101–106).

Singh, S. R. (2007a). A simple method of forecasting based on fuzzy time series. *Applied Mathematics and Computation, 186*, 330–339.

Singh, S. R. (2007b). A robust method of forecasting based on fuzzy time series. *Applied Mathematics and Computation, 188*, 472–484.

Singh, S. R. (2009). A computational method of forecasting based on fuzzy time series. *Mathematics and Computers in Simulation, 79*, 539–554.

Song, Q., & Chissom, B. S. (1993a). Fuzzy time series and its models. *Fuzzy Sets and Systems, 54*, 269–277.

Song, Q., & Chissom, B. S. (1993b). Forecasting enrollments with fuzzy time series – Part I. *Fuzzy Sets and Systems, 54*, 1–9.

Song, Q., & Chissom, B. S. (1994). Forecasting enrollments with fuzzy time series – Part II. *Fuzzy Sets and Systems, 62*, 1–8.

Tsai, C. C., & Wu, S. J. (2000). Forecasting enrolments with high-order fuzzy time series. In *Fuzzy Information Processing Society (NAFIPS), 19th international conference of the North American* (pp. 196–200).

Wang, N. Y., & Chen, S. M. (2009). Temperature prediction and TAIFEX forecasting based on automatic clustering techniques and two-factors high-order fuzzy time series. *Expert Systems with Applications, 36*, 2143–2154.

Yu, T. H. K., & Huarng, K. H. (2008). A bivariate fuzzy time series model to forecast the TAIEX. *Expert Systems with Applications, 34*, 2945–2952.

Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*, 338–353.