Granular Computing (2023) 8:373–390
https://doi.org/10.1007/s41066-022-00331-4

ORIGINAL PAPER

Adaptive hybrid fuzzy time series forecasting technique based on particle swarm optimization

Gunjan Goyal[1] · Dinesh C. S. Bisht[1]

Received: 13 January 2022 / Accepted: 19 March 2022 / Published online: 20 July 2022
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2022

Abstract

Fuzzy time series is a dynamic process in time series forecasting due to which it has gained a lot of attention from researchers. In this process, prediction accuracy is influenced by factors such as defining and partitioning the universe of discourse, fuzzification, and construction of the rule base, forecasting and defuzzification. Although numerous research have been provided in the literature, choosing the order of fuzzy time series and interval length is still a challenging task. This paper presents a computational forecasting model that overcomes the hassle of searching for the appropriate interval length and order of fuzzy time series. Particle swarm optimization is employed to search for the optimum interval length for the partitioning of the universe of discourse. Also, how changing its parameters affects the forecasting process is being investigated, which has never been done previously. A dynamic order approach is used for the selection of the order of fuzzy time series in the proposed model. In the proposed model, a sequence of orders is obtained in the training phase based upon forecast accuracy and then it is used for forecasting based upon certain rules. The model is tested on different actual time series, which include the benchmark data set of enrolments of Alabama University, the Taiwan stock exchange capitalization weighted stock index and also West Texas Intermediate crude oil prices. Different frequency datasets (e.g., yearly, monthly and daily) have been selected for this paper to check the robustness of the model. The root-mean-squared error is used as a performance parameter for the comparison of forecasting accuracy. The experimental results show that the proposed model performs better than the existing models in terms of forecasting accuracy.

Keywords Fuzzy time series · Dynamic order · Fuzzification · Particle swarm optimization · Forecasting

1 Introduction

Time series forecasting is the process of future observation prediction through the critical analysis of historical data. To predict future observations, the models are built and observations are made on historical data. There are many existing classical forecasting models such as linear regression, exponential smoothing, moving average models which work on basic assumption that the time series is stationary. But in real life situations, time series are complex in nature due to the fact that at times there is uncertainty in data. This uncertainty in data could also be possible due to inaccuracy in data. To handle the uncertainty in data, fuzzy set theory is an effective tool. A fuzzy set theory was proposed by Zadeh (1965) in which the linguistic terms are used for uncertain observations. Later, this theory was utilized in time series forecasting and termed as fuzzy time series (FTS). FTS forecasting models are dynamic in nature due to their rule-based structure.

Song and Chissom (1993a) proposed the concept of FTS with both the time-invariant and time-variant models and applied it to the enrollment of University of Alabama. The model proposed by Song and Chissom (1993b, 1994) uses the max–min operation that is computationally expensive. Then, Chen (1996) proposed a model using simple arithmetic operations that became popular among researchers. Since then a lot of significant work has been done towards the improvement of forecasting accuracy. The following are the major steps in the process of modeling fuzzy time series forecasting: (1) Defining and partitioning of Universe of Discourse (UoD), (2) Process of fuzzification, (3)

✉ Dinesh C. S. Bisht
drbisht.math@gmail.com

[1] Department of Mathematics, Jaypee Institute of Information Technology, Noida, UP, India

Rule base construction and (4) Forecasting and defuzzification process. Numerous novel models have been proposed and tested till now in diverse problem domains such as enrollment (Bisht and Kumar 2021), crop production (Singh and Borah 2014), stock market (Goyal and Bisht 2021), load forecasting (Sadaei et al. 2019) and shipping market time series (Gao et al. 2021).

Various models have been proposed in which different partitioning techniques are employed. In the partitioning process, the length of the interval is an important parameter. Some researchers have considered equal lengths of intervals while others have taken unequal lengths of intervals. Initially, Song and Chissom (1993b, 1994) and Chen (1996, 2002) gave the forecasting model using equal lengths of intervals. However, the impact of interval length was investigated by Huarng (2001). He gave the distribution and average based techniques whereas Huarng and Yu (2006) gave ratio-based technique to obtain length of interval. In the process of partitioning, along with arithmetic approaches, evolutionary algorithms have also been used to optimize the length (Lee et al. 2008; Kuo et al. 2009; Eğrioğlu 2012; Duru and Bulut 2014; Egrioglu et al. 2019; Zeng et al. 2019). The most commonly used evolutionary algorithms are genetic algorithms (GA) and particle swarm optimization (PSO). Many authors applied them in the fuzzy time series forecasting model in different processes such as partitioning of UoD, fuzzification of data points and construction of the rule base (Chen and Wang 2010; Aladag et al. 2012; Chen and Jian 2017; Chen and Phuong 2017; Chen et al. 2019; Pant and Kumar 2021a, b). Eğrioğlu (2012) proposed a time-invariant model that used both FCM and GA, whereas Aladag et al. (2012) proposed a time-invariant model that used both FCM and PSO. When the results of these two models are compared, it is clear that the model with PSO outperforms the model with GA. But for respective evolutionary algorithms, how parameters affect the FTS model has not been investigated. For this study, we have focused on PSO for the following reasons: PSO has an in-built guidance strategy, which results in faster convergence of PSO solutions, whereas GA does not have such a mechanism. PSO stores the previous best solutions obtained by each particle in memory, making the algorithm very robust. GA, on the other hand, does not use memory to keep track of solutions across generations. In process of fuzzification, the crisp values are converted to fuzzy sets. The impact of different fuzzy sets in fuzzy time series modeling has been observed by Bisht and Kumar (2016), Egrioglu et al. (2019) and Guan et al. (2019). Some researchers employed clustering algorithms to find clusters along with membership of each data point (Li et al. 2008; Askari and Montazerin 2015). Rule base is determined to identify the pattern of the time series. Different structures have been proposed in the literature for the identification of fuzzy logical relationships (FLR). Song and Chissom (1993b, 1994) gave the rule base in matrix form whereas Markov transition matrix was employed by Sullivan and Woodall (1994). Artificial neural network (ANN) and its variants have been used for the identification of FLR in the FTS model (Huarng and Yu 2006; Aladag et al. 2009; Singh 2018). While determining the rule base, the order of FTS plays an important role. The order of FTS boosts the accuracy of the model. This was tested by Chen (2002) in his extended work of Chen (1996). Thereafter, multiple researchers have used fixed high-order fuzzy time series to improve the model's accuracy (Aladag et al. 2009; Egrioglu et al. 2010; Panigrahi and Behera 2020). Generally, it was found that higher the order, better is the accuracy. But then arises the issue of over fitting on a training data set. The most common approach used for the process of defuzzification, i.e., reverse process fuzzification, is centroid method (Chen 1996; Huarng 2001).

From the above study, it has been observed that very less work is done in time-variant FTS models and appropriate partitioning technique along with order of FTS is still a challenging task. Also, for hybrid FTS models, how parameters of evolutionary algorithm affects FTS model has not been investigated. The present study presents a model that accommodates the following research objectives:

1. Partitioning technique and selection of appropriate order
2. Efficiency of model and
3. Optimal forecasting error.

A computationally robust model is proposed by combining particle swarm optimization and dynamic order algorithm. For partitioning of universe of discourse, particle swarm optimization is used as it has fewer parameters when compared with other nature inspired optimizations. Our study is about the effect of parameters of PSO when applied in fuzzy time series for optimization of partitioning of universe of discourse. Also, study on its parameters is done when PSO is combined with FTS model. And for the selection of an appropriate order, a dynamic order algorithm is used to auto-adjust the order of FTS model (Wagner et al. 2007). Also, proposed model minimizes the search for suitable defuzzification process.

The organization of the paper is done in the following ways: Basic definitions of FTS are defined in Sect. 2. Sections 3 and 4 describe particle swarm optimization and dynamic order algorithms, respectively. The proposed model is described in detail in Sect. 5. In Sect. 6, the model is tested on different frequency data sets and empirical study is presented in it. Finally, Sect. 7 is the conclusion.

2 Fuzzy time series

FTS is the concept proposed by Song and Chissom (1993a, b, 1994) based on fuzzy theory for the forecasting of time series. It can handle the forecasting of linguistic variable problems. The basic definitions are discussed below:

Definition 1 Let U be the UoD, where $U = \{\mathrm{u}_1, \mathrm{u}_2, \ldots, \mathrm{u}_\mathrm{n}\}$ on which fuzzy sets are defined as

$$A_j = \frac{\mu_{A_j}(\mathrm{u}_1)}{\mathrm{u}_1} + \frac{\mu_{A_j}(\mathrm{u}_2)}{\mathrm{u}_2} + \cdots \frac{\mu_{A_j}(\mathrm{u}_n)}{\mathrm{u}_n}, \tag{1}$$

where, $\mu_{A_j}(\mathrm{u}_n) \in (0, 1)$ is membership degree of u_n in A_j. Then the collection of fuzzy sets A_j is known as FTS on U, represented by $F(t)$.

Definition 2 Let $F(t)$ be the FTS and $\Re$ $(t, t-1)$ be a fuzzy relation, then $F(t) = F(t-1) \bigcirc \Re\ (t, t-1)$ where '$\bigcirc$' is an operator means $F(t)$ is caused by $F(t-1)$, represented by $F(t-1) \rightarrow F(t)$. It is known as first-order FTS and if $F(t)$ is caused by $F(t-1)$, $F(t-2)$, …, $F(t-m)$ $m > 0$, it is known as mth-order FTS which is represented by $F(t-1)$, $F(t-2)$, …, $F(t-m) \rightarrow F(t)$.

Definition 3 If for any time t, the fuzzy relation $\Re$ $(t, t-1)$ or $\Re$ $(t, t-m)$ is independent of t, then $F(t)$ is termed as the time-invariant FTS, else time-variant FTS. Here, independent of t means at different times t_1 and t_2, $\Re$ $(t_1, t_1-1) = \Re$ (t_2, t_2-1) or $\Re$ $(t_1, t_1-m) = \Re$ (t_2, t_2-m).

Definition 4 Let $F(t)$ be time-variant FTS, then relation is expressed as $F(t) = F(t-1) \bigcirc \Re^d\ (t, t-1)$, where d is the order of the FTS model which affects the forecast.

3 Particle swarm optimization

Kennedy and Eberhart (1995) proposed an optimization algorithm that mimics the swarm behavior known as particle swarm optimization (PSO). This optimization technique has an advantage over other nature inspired optimization techniques as it has few parameters, which makes it easy to implement. Also, in the process, no assumptions are required to handle the specific task, and it is computationally less expensive. PSO is an iterative process that uses the velocity displacement model to optimize a problem.

Initially in the algorithm, particles (solutions) are randomly generated with randomized velocity within the search space. The velocity and position of these particles are then updated in each iteration, keeping track of both the global and local best solutions until the termination criteria are met. These solutions are updated according to the following two equations:

$$v_i(t+1) = w * v_i(t+1) + c_1 r_1 (p_i^l(t) - p_i(t)) + c_2 r_2 (p_i^g(t) - p_i(t)), \tag{2}$$

$$p_i(t+1) = p_i(t) + v_i(t+1), \tag{3}$$

where w is the inertia weight, c_1 is the cognitive coefficient and c_2 is the social coefficient, r_1 and r_2 are randomly generated numbers in the range [0, 1]. In Eq. (2), $c_1 r_1 \left(p_i^l(t) - p_i(t)\right)$ is the personal influence and $c_2 r_2 (p_i^g(t) - p_i(t))$ is the social influence. The algorithm works with static values c_1 and c_2 with their sum equal to 4. Since c_1 and c_2 determine the inclination of search, a higher value of c_1 means greater local search ability, whereas a higher c_2 means greater global search ability. So, they are generally assumed to be equal in keeping away divergence and cyclic behavior. To avoid the quick convergence of solutions, r_1 and r_2 are used. Premature convergence may occur if the inertia weight is chosen incorrectly, as its role is to explore and exploit. The value of w is problem dependent and lies between 0 and 1.

4 Dynamic order algorithm

This paper proposes a dynamic order PSO-based fuzzy time series model to auto-adjust the order of fuzzy time series and partition the UoD. The selection of the order is important to any forecasting model's success. When the forecasting problem is not well understood, automatic determination of this window size is essential. The dynamic order algorithm automatically adjusts its window size with each slide of the window. The dynamic approach used in this paper to select the appropriate order was proposed by Wong et al. (2010). In each round of the training phase, the order is adjusted dynamically. This is done in the following manner.

1. Initialized by taking $i = 1$ and $n = 1$.
2. Then, two orders are selected as n and $n + i$, and the flag $\mathrm{h} = n$, where both n and i are positive integers.

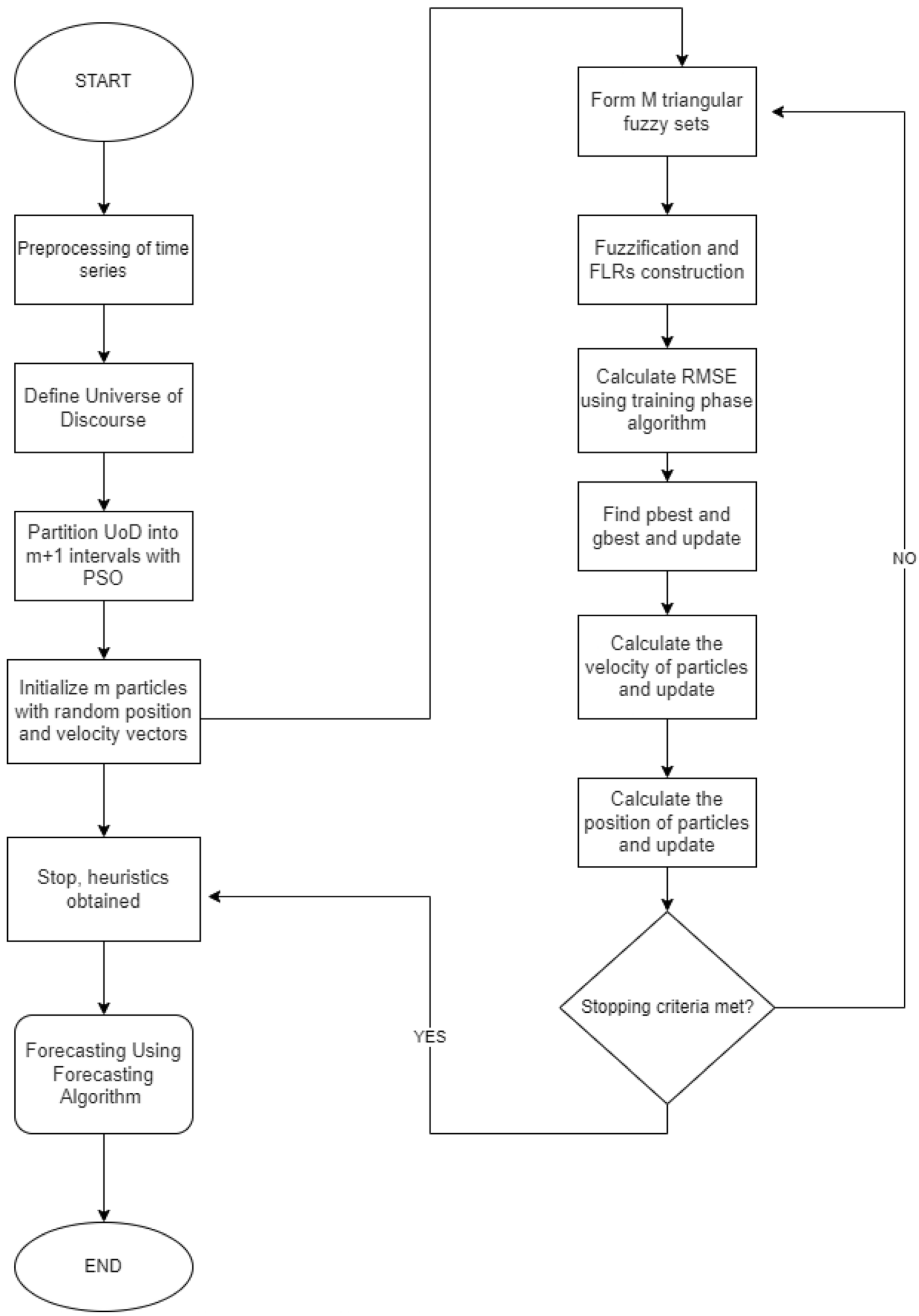


Fig. 1 Flowchart of proposed model

3. With the selected two orders, the next data point is forecast using the proposed model. Both orders' predictive accuracy (PA) is computed using Eq. (4) (PA_n and PA_{n+i}). Best solution among these two will be selected to predict next order and flag.

$$\text{Predictive Accuracy} = |\text{actual value} - \text{forecasted value}|. \quad (4)$$

4. Now select two more window sizes based on which one had the best accuracy. If $PA_n \leq PA_{n+i}$, two new orders are created: n and $n-i$, with flag h equal to $n-i$. If $n=0$, then the process starts from the initial step. If $PA_n < PA_{n+i}$, two new orders are created: $n+i$ and $n+2i$, with flag h equal to $n+i$.
5. To include the next time series observation, slide the h to the right. Use the two new orders obtained to run

two more dynamic generations, predict future data, and assess their accuracy.

6. This process is repeated till the sequence of flags is obtained for all historical data.

5 Proposed model

In this paper, a computational FTS model is presented in which PSO is used to search for the optimal partitioning of UoD and a dynamic order approach has been used for the order selection of FTS. PSO is employed to optimize the length of interval by determining the boundary points of interval because optimization of interval length has a great impact on the fuzzification process and the accuracy of results. The problem of selecting an appropriate order for the model is resolved using a dynamic order approach in which a sequence of orders is obtained in training algorithm and then in the forecasting algorithm, the order is selected from this sequence based on certain rules. The methodology consists of two phases which involve pre-processing, defining and partitioning of the UoD, fuzzification, and construction of the rule base, forecasting and defuzzification. The steps of the proposed model are explained below and also explained using flowchart (Fig. 1).

Phase 1: Pre-processing, defining and partitioning of UoD, fuzzification, construction of rule base

Step 1 The process is initiated by checking the outliers in the time series using the generalized extreme studentized deviate (ESD) test, an extension of the Grubbs test (Grubbs 1950). And the outliers found are replaced using linear interpolation.

Step 2 Define UoD based on range values of the data series defined by U,

$$U = \left[X_{\text{lb}}, X^{\text{ub}}\right], \tag{5}$$

where $X_{\text{lb}} = X_{\min} - \mu$, $X^{\text{ub}} = X_{\max} + \mu$ and $\mu = \frac{\sum_{i=1}^{n} |X_{i+1} - X_i|}{n-1}$, n is the total number of data points.

Step 3 U is now partitioned into $m+1$ intervals: $i_1, i_2, \ldots, i_{m+1}$ using PSO. The optimal partition vector $P = [p_1, p_2, p_3, \ldots, p_m]$ is obtained to get the optimal partition of intervals $i_1, i_2, \ldots, i_{m+1}$ where $i_1 = [X_{\text{lb}}, p_1]$, $i_2 = [p_1, p_2]$ $i_3 = [p_2, p_3]$, …, $i_{m+1} = [p_m, X^{\text{ub}}]$ from the following procedure. The parameter values are taken as $c_1 = c_2 = 2$, inertia weight is varied in range 0–1 and the fitness function is root-mean-squared error (RMSE).

Step 3.1 Initially, m particles are generated with position vectors and velocity vectors randomly.

Step 3.2 Calculate the RMSE of each particle.

Step 3.3 Next, if RMSE of each particle's current position is better than its personal best position vector, then personal best position is updated.

Step 3.4 Particle having least RMSE is chosen as the best particle.

Step 3.5 Now the elements of velocity vector are updated based on Eq. (2) and position vector's elements are updated based on equation Eq. (3).

Step 3.6 The process from steps 3.2 to 3.5 is repeated until the termination criteria are satisfied. Here, the process is terminated when the number of iteration reaches the predefined value.

Step 4 Once m particles are obtained, m+1 intervals are formed and based on it, m triangular fuzzy sets are defined.

Step 5 Fuzzify the data and establish the fuzzy logical relation between time t and $t+1$ as $A_i \rightarrow A_j$ in training phase and $A_i \rightarrow \#$ in forecasting phase.

Phase 2: Forecasting and defuzzification

Step 6 Now in this step, we have training phase and forecasting phase algorithm.

Step 6.1 In training phase, for each data, two orders are determined using dynamic order algorithm and then forecasted values for those two orders are determined with the help of training algorithm. Based on forecasting accuracy, the one with higher accuracy is selected. Dynamic order algorithm is initialized by taking $i = 1$ and $n = 1$.

Following are the notations used in both training and forecasting algorithms:

I_k represents kth interval
$X(t)$ is the actual data value at time t
$[{}^{l}A_k]$ defines the lower bound of interval I_k
$[{}^{m}A_k]$ defines the middle value of interval I_k
$[{}^{u}A_k]$ defines the upper bound of interval I_k
$Y(t+1)$ is forecasted value at time $t+1$
h is the dynamic order
c is count; s is sum and d is deviation
${}^{+}R_N$, ${}^{-}R_N$, ${}^{+}P_N$, ${}^{-}P_N$ are fuzzy predictors
p is number of steps to be computed.

Training algorithm: When FLR between time t and t+1: $A_i \rightarrow A_j$

c = 0, s= 0, d= 0;
if h = 1,
    $Y(t+1) = [{}^{m}A_j]$
if h > 1,
    for i = 0,1,2… h -2
        $d = d + \left(X(t-i) - X(t-i-1)\right)^2$
        $d = \sqrt{\frac{d}{h-1}}$
    end for loop
    p= ceil (number of intervals/2)
    for N = 1, 2, 3..., p
        ${}^{+}R_N = X(t) + d * N$, ${}^{-}R_N = X(t) - d * N$
        ${}^{+}P_N = X(t) + d/2N$, ${}^{-}P_N = X(t) - d/2N$
        if ${}^{+}R_N \geq [{}^{l}A_j]$ and ${}^{+}R_N \leq [{}^{u}A_j]$, then $s = s + {}^{+}R_N$ and c = c + 1
        if ${}^{-}R_N \geq [{}^{l}A_j]$ and ${}^{-}R_N \leq [{}^{u}A_j]$, then $s = s + {}^{-}R_N$ and c = c + 1
        if ${}^{+}P_N \geq [{}^{l}A_j]$ and ${}^{+}P_N \leq [{}^{u}A_j]$, then $s = s + {}^{+}P_N$ and c = c + 1
        if ${}^{-}P_N \geq [{}^{l}A_j]$ and ${}^{-}P_N \leq [{}^{u}A_j]$, then $s = s + {}^{-}P_N$ and c = c + 1
    end for loop
    If $[{}^{l}A_j] < s$ and $s < [{}^{m}A_j]$ and $(s - [{}^{l}Aj]) < (s - [{}^{m}A_j])$
        then $Y(t+1) = (s + [{}^{l}A_j]) / (c+1)$
    If $[{}^{m}A_j] < s$ and $s < [{}^{u}A_j]$ and $(s - [{}^{u}A_j]) < (s - [{}^{m}A_j])$
        then $Y(t+1) = (s + [{}^{u}A_j]) / (c+1)$
    else
        $Y(t+1) = (s + [{}^{m}A_j]) / (c+1)$

Step 6.2 From the training phase algorithm, the sequence of order is obtained reflecting the trend of prediction. In forecasting phase, the forecasting algorithm forecasts two values, where the order and forecasted value is determined by undermentioned rules, considering sequence of flags.

1. If $h_t = 1$, then order selected for time $t + 1$ is 1 and 2.
2. If $h_t = k$ and $h_t \geq h_{t-1}$, then order selected for time $t + 1$ is k and $k + 1$.
3. If $h_t = k$ and $h_t \leq h_{t-1}$, then order selected for time $t + 1$ is k and $k - 1$.
4. If $h_t \geq h_{t-1}$ and $X_t \geq X_{t-1}$, then Y_{t+1} is max of two forecasted values.
5. If $h_t < h_{t-1}$ and $X_t < X_{t-1}$, then Y_{t+1} is min of two forecasted values.
6. If $h_t \geq h_{t-1}$ and $X_t \leq X_{t-1}$ or $h_t \leq h_{t-1}$ and $X_t \geq X_{t-1}$, then Y_{t+1} is mean of two forecasted values.

Forecasting algorithm: When FLR between time t and t+1: $A_i \rightarrow \#$

c = 0, s= 0, d= 0;
if ħ = 1,
    then Y(t+1) = [_Aj_]
if ħ > 1,
    for i = 0, 1, 2…, ħ -2
        $d = d + \left(X(t-i) - X(t-i-1)\right)^2$
        $d = \sqrt{\frac{d}{ħ-1}}$
    end for loop
    p= ceil (number of intervals/2)
    for N = 1,2,3...p
        ${}^{+}R_N = X(t) + d*N,\ {}^{-}R_N = X(t) - d*N$
        ${}^{+}P_N = X(t) + d/2N,\ {}^{-}P_N = X(t) - d/2N$
        if ${}^{+}R_N \geq [{}^{l}A_j]$ and ${}^{+}R_N \leq [{}^{u}A_j]$, then s = s + ${}^{+}R_N$ and c = c + 1
        if ${}^{-}R_N \geq [{}^{l}A_j]$ and ${}^{-}R_N \leq [{}^{u}A_j]$, then s = s + ${}^{-}R_N$ and c = c + 1
        if ${}^{+}P_N \geq [{}^{l}A_j]$ and ${}^{+}P_N \leq [{}^{u}A_j]$, then s = s + ${}^{+}P_N$ and c = c + 1
        if ${}^{-}P_N \geq [{}^{l}A_j]$ and ${}^{-}P_N \leq [{}^{u}A_j]$, then s = s + ${}^{-}P_N$ and c = c + 1
    end for loop
    Y(t+1) = $(s + [{}^{m}A_j]) / (c+1)$

6 Empirical study

In this paper, the model is tested on seven datasets which includes the benchmark data set of enrolments of Alabama University and the Taiwan stock exchange capitalization weighted stock index (TAIEX) and also West Texas Intermediate (WTI) crude oil prices. The data set of enrolments of Alabama University is yearly data whereas dataset of TAIEX is daily dataset and WTI crude oil prices dataset is monthly dataset. The descriptive statistics of each dataset is briefly described in Table 1. The comparison of proposed FTS model with the existing models is done in terms of RMSE. The benchmark FTS models selected for comparison are exponentially weighted FTS (Sadaei et al. 2014), improved weighted FTS (Efendi et al. 2013), conventional FTS (Chen 1996), trend-weighted FTS (Cheng et al. 2009), and weighted FTS (Yu 2005). Moreover, mean fitness value is considered over 30 runs of each experiment to check the robustness of the model. Also, the number of intervals and inertial weight is varied in the experiment. To the best of our knowledge, there is no fixed method of selecting the number of fuzzy sets. The cognitive constant and social constant are set to 2, based on the literature. Table 2 describes the parameters setting of the proposed model. Further, Fig. 2 illustrates the graph between fitness obtained by proposed model and number of iterations. It is observed from the figure that it is decreasing but not linearly which means model is achieving better solution precision.

Table 1 Characteristics of all datasets

Datasets	Maximum	Minimum	Median	Mean	STD	Skewness	Kurtosis
Enrollment	19,337.00	13,055.00	15,732.00	16,194.18	1816.49	0.36	− 0.54
TAIEX 1999	8608.91	5474.79	7590.43	7408.65	739.46	− 0.76	− 0.21
TAIEX 2000	10,202.20	4614.62	8390.33	7904.31	1602.79	− 0.56	− 0.96
TAIEX 2001	6104.24	3446.26	5057.06	4911.54	701.76	− 0.30	− 1.01
TAIEX 2002	6462.29	3850.04	5316.04	5246.58	663.45	− 0.04	− 1.18
TAIEX 2003	6142.31	4139.5	5095.24	5151.75	607.87	− 0.005	− 1.43
TAIEX 2004	7034.10	5316.87	5936.53	6032.27	412.84	0.52	− 0.43
WTI crude oil	133.88	11.35	30.71	44.05	29.26	0.87	− 0.37

Table 2 Parameter setting of the proposed model

Parameter	Value
Number of runs	30
Population size	50
Number of iterations	100
Cognitive constant (c_1)	2
Social constant (c_2)	2
Inertia weight (w)	0.2, 0.4, 0.6, 0.8, 1
Number of fuzzy sets	7, 8, 9, 10

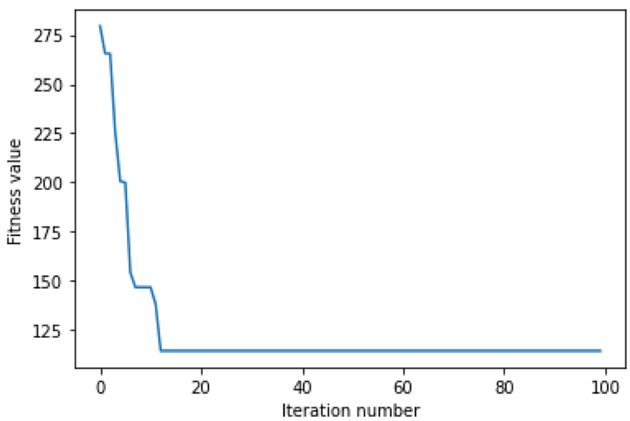


Fig. 2 Fitness over iterations

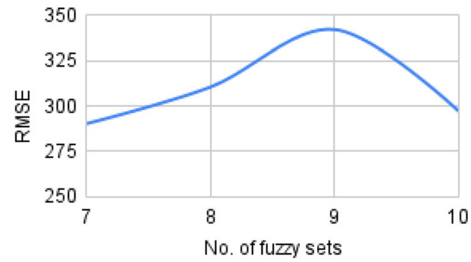


Fig. 4 RMSE vs number of fuzzy sets of enrollment dataset of University of Alabama

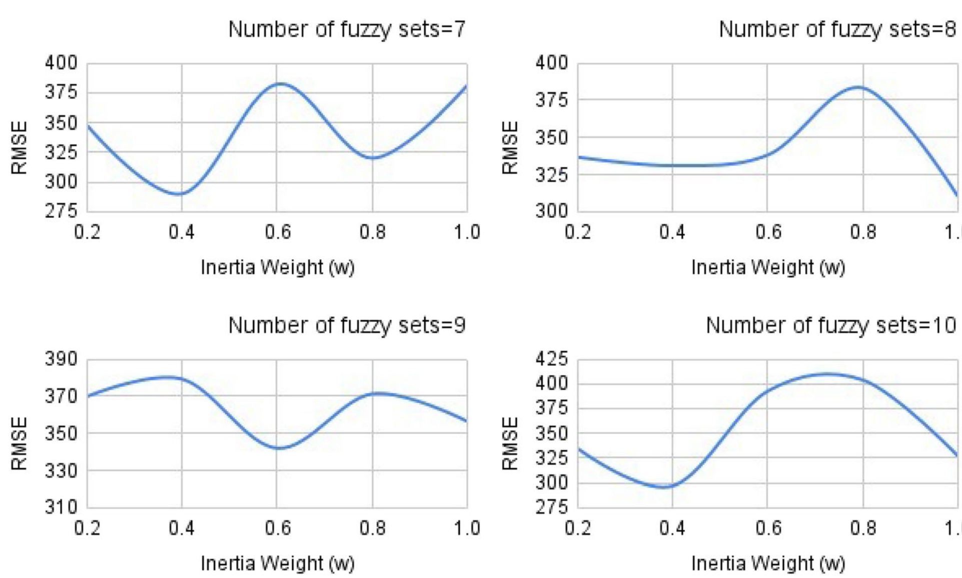
Table 3 Comparison of the RMSE of enrollment dataset of University of Alabama for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	326.92	259.33	224.41	215.25
Cheng et al. (2009)	326.92	259.33	224.41	215.25
Sadaei et al. (2014)	326.92	259.33	224.41	215.25
Chen (1996)	326.92	259.33	224.41	215.25
Efendi et al. (2013)	326.92	259.33	224.41	215.25
Proposed model	290.15	310.37	341.96	297.00

6.1 Enrolments of Alabama University

The benchmark dataset of enrolments of Alabama University is a yearly data from 1971 to 1992. The data are divided into 2 parts: 80% training and 20% testing. In Fig. 3, graphs represent the effect of w on RMSE for different numbers of fuzzy sets on test data whereas Fig. 4 shows the varying RMSE when number of fuzzy sets are changing. The comparative results of RMSE on test data are shown in Table 3. It is observed that results from proposed model are better when number of fuzzy sets are 7. Also, variation in RMSE is very low when number of fuzzy sets are varied from proposed model as compared to existing models.

Fig. 3 RMSE vs inertia weight (w) for different number of fuzzy sets of enrollment dataset of University of Alabama



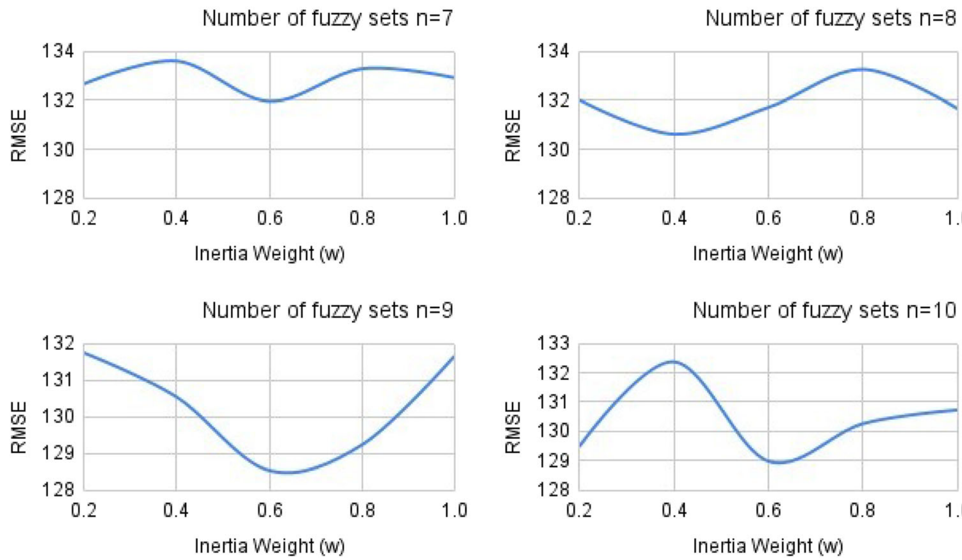


Fig. 5 RMSE vs inertia weight (w) for different number of fuzzy sets of TAIEX 1999

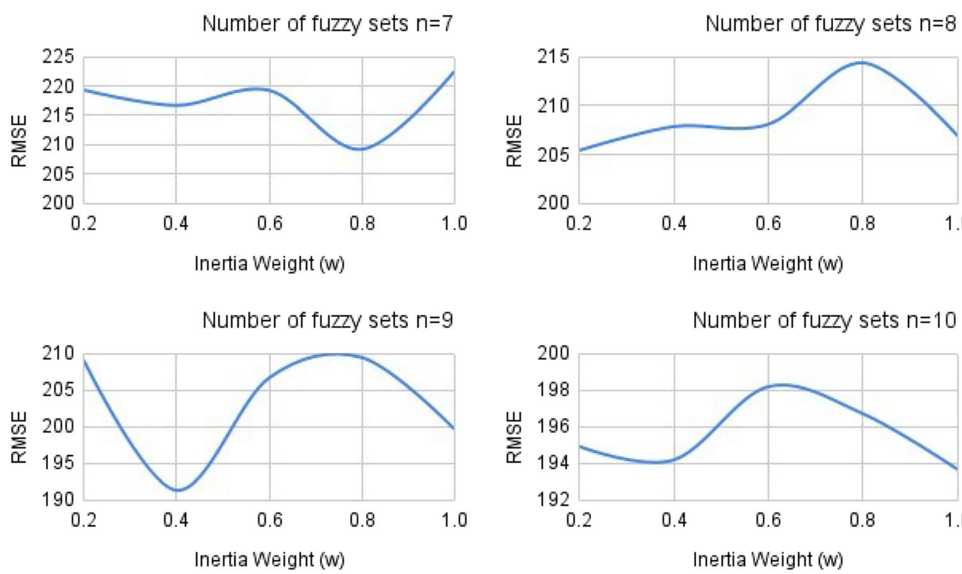


Fig. 6 RMSE vs inertia weight (w) for different number of fuzzy sets of TAIEX 2000

6.2 Taiwan stock exchange capitalization weighted stock index (TAIEX)

The proposed model is applied to TAIEX from year 1999 to 2004. Further, it is divided in 2 parts: 80% training and 20% testing for each year. In Figs. 5, 6, 7, 8, 9 and 10, graphs represent the effect of w on RMSE for different number of fuzzy sets on test data whereas Fig. 11 shows the varying RMSE when number of fuzzy sets are changing. Figures 5 and 6 are graphs representing the effect of w on RMSE for different number of fuzzy sets in forecasting TAIEX 1999 and TAIEX 2000. Figures 7 and 8 are graphs representing the effect of w on RMSE for different number of fuzzy sets in forecasting TAIEX 2001 and TAIEX 2002. Figures 9 and 10 are graphs representing the effect of w on RMSE for different number of fuzzy sets in forecasting TAIEX 2003 and TAIEX 2004.

The comparative results of RMSE on test data for different number of fuzzy sets for TAIEX 1999 is shown in Table 4, for TAIEX 2000 is shown in Table 5, for TAIEX 2001 is shown in Table 6, for TAIEX 2002 is shown in Table 7, for TAIEX 2003 is shown in Table 8, for TAIEX 2004 is shown in Table 9. It is observed that results from proposed model are better than the existing models. Also, when number of fuzzy sets changes, there is less variation in RMSE from proposed model as compared to existing models. The comparison between the actual and forecasted data of TAIEX 1999 and TAIEX 2000 is shown in Figs. 12 and 13. The comparison between the actual and forecasted data of TAIEX 2001 and TAIEX 2002 is shown in Figs. 14

Fig. 7 RMSE vs inertia weight (w) for different number of fuzzy sets of TAIEX 2001

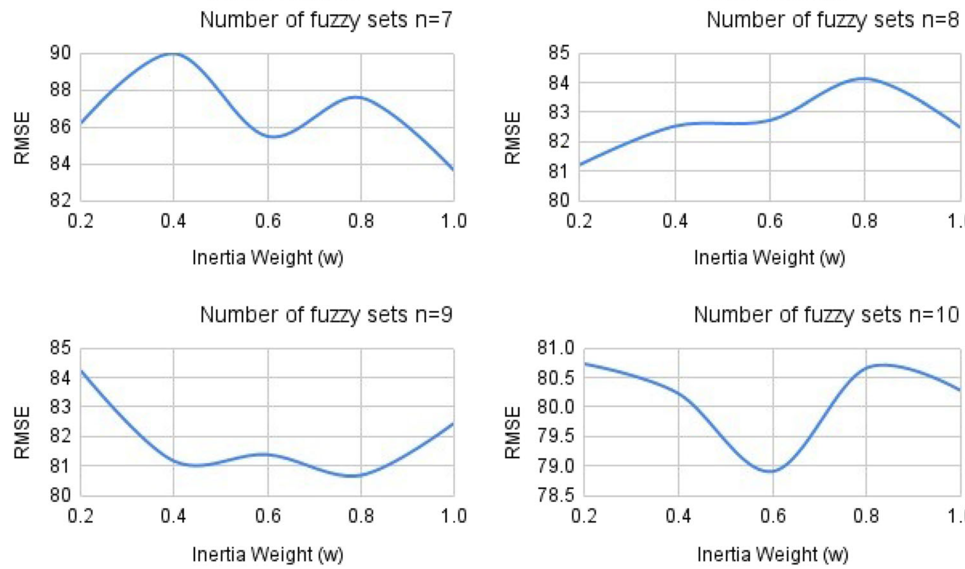


Fig. 8 RMSE vs inertia weight (w) for different number of fuzzy sets of TAIEX 2002

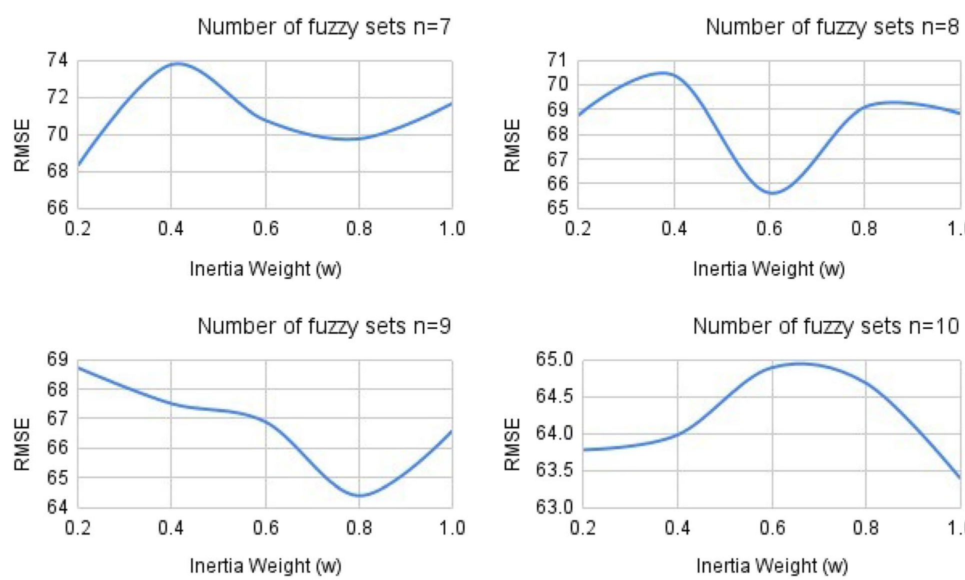


Fig. 9 RMSE vs inertia weight (w) for different number of fuzzy sets of TAIEX 2003

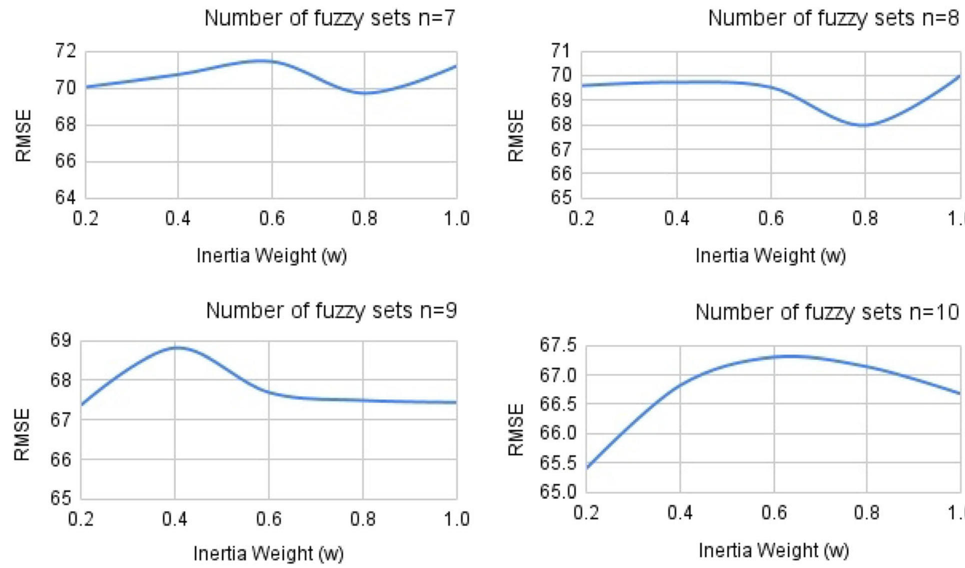


Fig. 10 RMSE vs inertia weight (w) for different number of fuzzy sets of TAIEX 2004

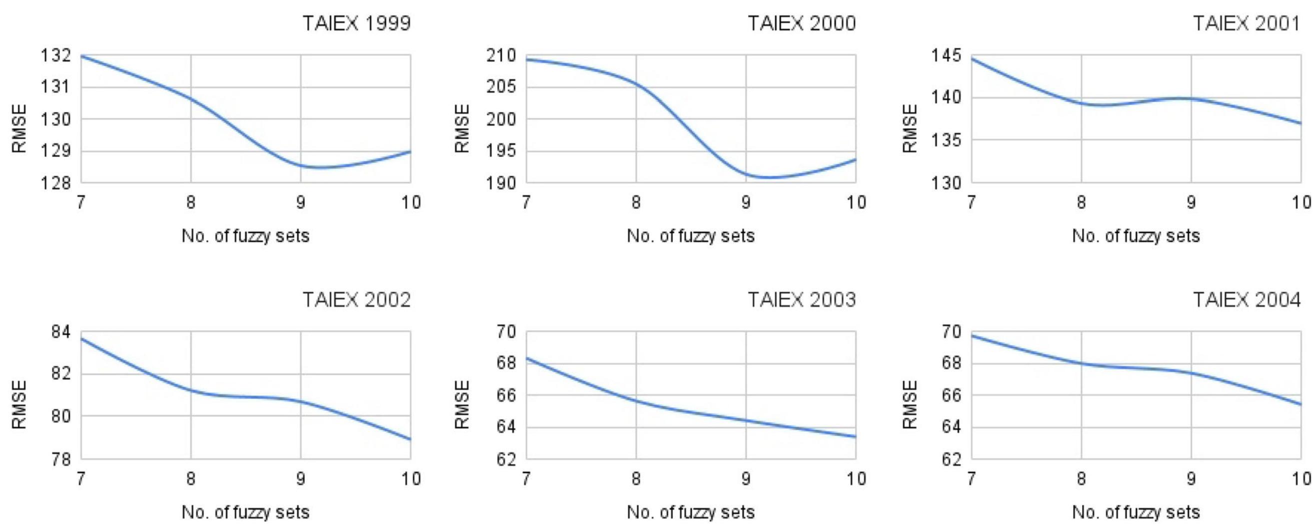


Fig. 11 RMSE vs number of fuzzy sets of TAIEX dataset

Table 4 Comparison of the RMSE of TAIEX 1999 test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	147.29	158.91	132.16	124.64
Cheng et al. (2009)	143.04	167.56	140.79	122.59
Sadaei et al. (2014)	149.67	156.32	128.59	125.95
Chen (1996)	181.69	190.17	159.85	146.51
Efendi et al. (2013)	147.72	164.86	137.79	124.54
Proposed model	131.95	130.61	128.53	128.98

Table 5 Comparison of the RMSE of TAIEX 2000 test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	250.03	245.70	225.29	239.03
Cheng et al. (2009)	253.43	238.16	229.02	241.98
Sadaei et al. (2014)	249.31	245.73	227.35	239.14
Chen (1996)	461.01	371.47	322.31	298.15
Efendi et al. (2013)	261.88	241.70	230.41	245.07
Proposed model	209.24	205.43	191.35	193.66

Table 6 Comparison of the RMSE of TAIEX 2001 test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	147.75	148.34	109.97	143.18
Cheng et al. (2009)	160.16	153.36	112.93	154.39
Sadaei et al. (2014)	135.73	142.07	106.74	136.33
Chen (1996)	160.82	165.34	117.79	147.64
Efendi et al. (2013)	156.00	150.39	112.26	149.08
Proposed model	144.55	139.32	139.83	136.97

Table 7 Comparison of the RMSE of TAIEX 2002 test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	92.16	85.71	88.39	73.18
Cheng et al. (2009)	116.73	88.96	102.19	84.52
Sadaei et al. (2014)	90.35	93.66	89.95	71.70
Chen (1996)	122.06	101.14	108.5	86.54
Efendi et al. (2013)	99.31	85.78	94.89	80.55
Proposed model	83.60	81.22	80.69	78.91

Table 8 Comparison of the RMSE of TAIEX 2003 test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	96.73	79.73	82.39	67.97
Cheng et al. (2009)	103.09	97.14	89.91	67.37
Sadaei et al. (2014)	94.31	74.68	81.46	72.99
Chen (1996)	144.75	102.01	107.8	85.6
Efendi et al. (2013)	100.08	89.32	89.7	68.25
Proposed model	68.32	65.63	64.41	63.39

Table 9 Comparison of the RMSE of TAIEX 2004 test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	70.12	70.62	62.07	72.84
Cheng et al. (2009)	74.1	75.78	67.75	78.53
Sadaei et al. (2014)	70.6	68.04	61.44	73.55
Chen (1996)	77.06	152.6	110.99	120.39
Efendi et al. (2013)	72.89	76.86	65.73	77.71
Proposed model	69.72	67.98	67.37	65.41

and 15.The comparison between the actual and forecasted data of TAIEX 2003 and TAIEX 2004 is shown in Figs. 16 and 17.

6.3 WTI crude oil prices

The dataset of Cushing, Oklahoma, U.S. is considered as the largest oil storage tank farm in the world. The amount of crude oil stored at Oklahoma, controls its price all over the world and because of that Cushing is pricing point for WTI oil prices. The data is collected from the site https://www.eia.gov/ from January 1986 to August 2019. It is divided into 2 parts: 80% training and 20% testing data. Figure 18 is the graph representing the effect of w on RMSE for different numbers of fuzzy sets on test data whereas Fig. 19 shows the varying RMSE when number of fuzzy sets are changing. The comparative results of RMSE on test data is shown in Table 10. It is observed that results from proposed model are better when number of fuzzy sets are 7 and 8. Also, variation in RMSE is less when number of fuzzy sets are varied from proposed model as compared to existing models. Figure 20 shows the comparison between the actual and forecasted data.

Fig. 12 Graphical comparison between actual and forecasted data of TAIEX 1999

Fig. 13 Graphical comparison between actual and forecasted data of TAIEX 2000

Fig. 14 Graphical comparison between actual and forecasted data of TAIEX 2001

Fig. 15 Graphical comparison between actual and forecasted data of TAIEX 2002

Fig. 16 Graphical comparison between actual and forecasted data of TAIEX 2003

Fig. 17 Graphical comparison between actual and forecasted data of TAIEX 2004

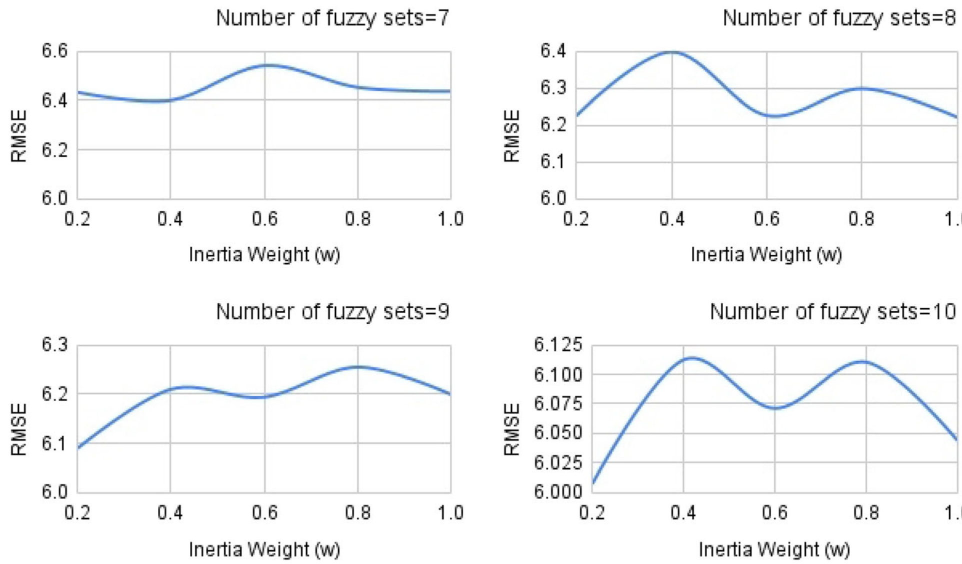


Fig. 18 RMSE vs inertia weight (w) for different number of fuzzy sets of WTI crude oil prices test data

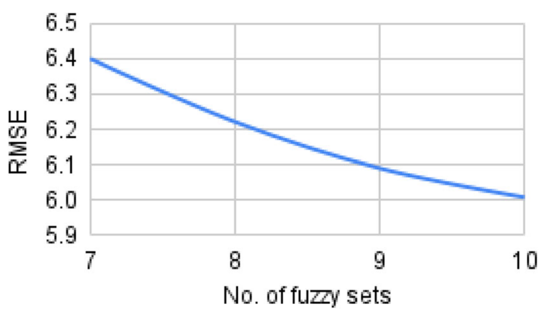


Fig. 19 RMSE vs number of fuzzy sets of WTI crude oil prices test data

Table 10 Comparison of the RMSE of WTI crude oil price test data for different numbers of fuzzy sets

Models	Number of fuzzy sets			
	7	8	9	10
Yu (2005)	7.31	7.48	6.22	6.23
Cheng et al. (2009)	7.03	6.67	5.84	6.05
Sadaei et al. (2014)	7.35	7.38	6.25	6.18
Chen (1996)	6.95	6.07	6.5	5.88
Efendi et al. (2013)	7.11	6.76	5.97	5.89
Proposed model	6.40	6.22	6.08	6.007

7 Conclusion

This study suggests a computational fuzzy time series model that combines particle swarm optimization and dynamic order algorithm. PSO was applied to optimize the interval length. The presented study is based on a dynamic order algorithm in which the order selection of FTS is adaptive and the hassle of the defuzzification process is reduced. The advantage of the proposed method is that it is adaptive in nature, optimal length of interval is obtained by PSO and provides the forecasted value in crisp form which reduces the need of defuzzification process. The model is tested and evaluated on enrolments of University of Alabama, stock indices of the Taiwan Stock Exchange, and Cushing, WTI spot prices of crude oil. The data were divided into 2 parts: 80% for training and 20% for testing. This study investigated the effects of inertia weight and the number of fuzzy sets on RMSE. The comparison of results is done on the basis of RMSE with the existing models (Yu 2005; Cheng et al. 2009; Sadaei et al. 2014; Chen 1996; Efendi et al. 2013), and it is observed that the proposed model performs better than the existing models. Graphs show how inertia weight and the number of fuzzy sets affect the fitness of FTS models. Also, less variation was observed in the results, which shows the robustness of the model. This work can be expanded by applying deep learning models to fuzzy time series and examining the effects of their parameters on the FTS model.

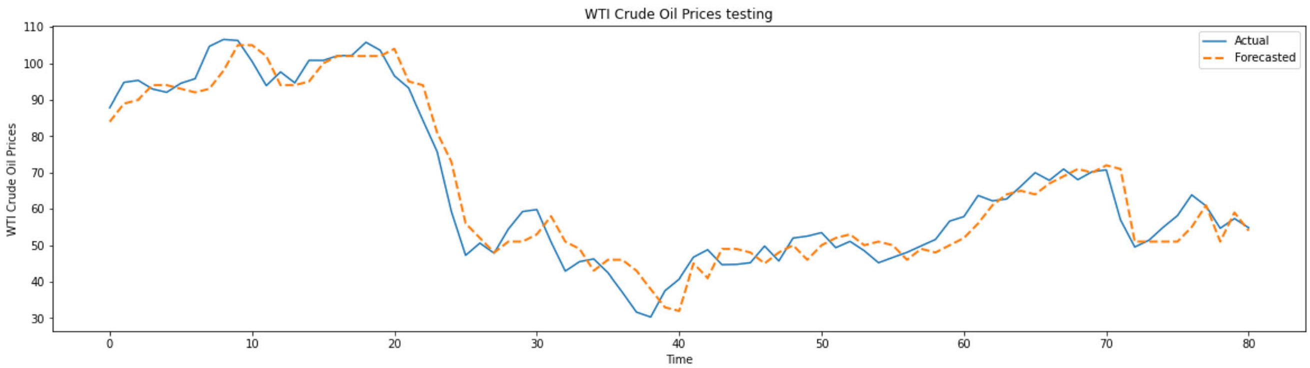


Fig. 20 Graphical comparison between actual and forecasted data of WTI crude oil prices

Data availability Statement Data will be made available on reasonable request.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

Aladag CH, Basaran MA, Egrioglu E et al (2009) Forecasting in high order fuzzy times series by using neural networks to define fuzzy relations. Expert Syst Appl 36:4228–4231

Aladag CH, Yolcu U, Egrioglu E, Dalar AZ (2012) A new time invariant fuzzy time series forecasting method based on particle swarm optimization. Appl Soft Comput 12:3291–3299

Askari S, Montazerin N (2015) A high-order multi-variable fuzzy time series forecasting algorithm based on fuzzy clustering. Expert Syst Appl 42:2121–2135

Bisht K, Kumar S (2016) Fuzzy time series forecasting method based on hesitant fuzzy sets. Expert Syst Appl 64:557–568

Bisht K, Kumar A (2021) A method for fuzzy time series forecasting based on interval index number and membership value using fuzzy c-means clustering. Evol Intell 1–13. https://doi.org/10.1007/s12065-021-00656-0

Chen S-M (1996) Forecasting enrollments based on fuzzy time series. Fuzzy Sets Syst 81:311–319

Chen S-M (2002) Forecasting enrollments based on high-order fuzzy time series. Cybern Syst 33:1–16

Chen S-M, Jian W-S (2017) Fuzzy forecasting based on two-factors second-order fuzzy-trend logical relationship groups, similarity measures and PSO techniques. Inf Sci 391:65–79

Chen S-M, Phuong BDH (2017) Fuzzy time series forecasting based on optimal partitions of intervals and optimal weighting vectors. Knowl Based Syst 118:204–216

Chen S-M, Wang N-Y (2010) Fuzzy forecasting based on fuzzy-trend logical relationship groups. IEEE Trans Syst Man Cybern Part B (cybern) 40:1343–1358

Chen S-M, Zou X-Y, Gunawan GC (2019) Fuzzy time series forecasting based on proportions of intervals and particle swarm optimization techniques. Inf Sci 500:127–139

Cheng C-H, Chen Y-S, Wu Y-L (2009) Forecasting innovation diffusion of products using trend-weighted fuzzy time-series model. Expert Syst Appl 36:1826–1832

Duru O, Bulut E (2014) A non-linear clustering method for fuzzy time series: histogram damping partition under the optimized cluster paradox. Appl Soft Comput 24:742–748

Efendi R, Ismail Z, Deris MM (2013) Improved weight Fuzzy Time Series as used in the exchange rates forecasting of US Dollar to Ringgit Malaysia. Int J Comput Intell Appl 12:1350005

Eğrioğlu E (2012) A new time-invariant fuzzy time series forecasting method based on genetic algorithm. Adv Fuzzy Syst 2012:785709

Egrioglu E, Aladag CH, Yolcu U et al (2010) Finding an optimal interval length in high order fuzzy time series. Expert Syst Appl 37:5052–5055

Egrioglu E, Yolcu U, Bas E (2019) Intuitionistic high-order fuzzy time series forecasting method based on pi-sigma artificial neural networks trained by artificial bee colony. Granul Comput 4:639–654

Gao R, Duru O, Yuen KF (2021) High-dimensional lag structure optimization of fuzzy time series. Expert Syst Appl 173:114698

Goyal G, Bisht DC (2021) Strong α-cut and associated membership-based modeling for fuzzy time series forecasting. Int J Model Simul Sci Comput 12:2050067

Grubbs FE (1950) Sample criteria for testing outlying observations. Ann Math Stat 21:27–58

Guan H, He J, Guan S, Zhao A (2019) Neutrosophic soft sets forecasting model for multi-attribute time series. IEEE Access 7:25575–25588

Huarng K (2001) Effective lengths of intervals to improve forecasting in fuzzy time series. Fuzzy Sets Syst 123:387–394

Huarng K, Yu TH-K (2006) Ratio-based lengths of intervals to improve fuzzy time series forecasting. IEEE Trans Syst Man Cybern Part B (cybern) 36:328–340

Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks. IEEE, pp 1942–1948

Kuo I-H, Horng S-J, Kao T-W et al (2009) An improved method for forecasting enrollments based on fuzzy time series and particle swarm optimization. Expert Syst Appl 36:6108–6117

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 Syst Appl 34:328–336

Li S-T, Cheng Y-C, Lin S-Y (2008) A FCM-based deterministic forecasting model for fuzzy time series. Comput Math Appl 56:3052–3063

Panigrahi S, Behera HS (2020) A study on leading machine learning techniques for high order fuzzy time series forecasting. Eng Appl Artif Intell 87:103245

Pant M, Kumar S (2021a) Fuzzy time series forecasting based on hesitant fuzzy sets, particle swarm optimization and support vector machine-based hybrid method. Granul Comput 1–19
Pant M, Kumar S (2021b) Particle swarm optimization and intuitionistic fuzzy set-based novel method for fuzzy time series forecasting. Granul Comput 7:1–19
Sadaei HJ, Enayatifar R, Abdullah AH, Gani A (2014) Short-term load forecasting using a hybrid model with a refined exponentially weighted fuzzy time series and an improved harmony search. Int J Electr Power Energy Syst 62:118–129
Sadaei HJ, de Lima e Silva PC, Guimarães FG, Lee MH (2019) Short-term load forecasting by using a combined method of convolutional neural networks and fuzzy time series. Energy 175:365–377
Singh P (2018) Rainfall and financial forecasting using fuzzy time series and neural networks based model. Int J Mach Learn Cybern 9:491–506
Singh P, Borah B (2014) An effective neural network and fuzzy time series-based hybridized model to handle forecasting problems of two factors. Knowl Inf Syst 38:669–690
Song Q, Chissom BS (1993a) Fuzzy time series and its models. Fuzzy Sets Syst 54:269–277
Song Q, Chissom BS (1993b) Forecasting enrollments with fuzzy time series—part I. Fuzzy Sets Syst 54:1–9
Song Q, Chissom BS (1994) Forecasting enrollments with fuzzy time series—part II. Fuzzy Sets Syst 62:1–8
Sullivan J, Woodall WH (1994) A comparison of fuzzy forecasting and Markov modeling. Fuzzy Sets Syst 64:279–293
Wagner N, Michalewicz Z, Khouja M, McGregor RR (2007) Time series forecasting for dynamic environments: the DyFor genetic program model. IEEE Trans Evol Comput 11:433–452
Wong W-K, Bai E, Chu AW-C (2010) Adaptive time-variant models for fuzzy-time-series forecasting. IEEE Trans Syst Man Cybern Part B (cybern) 40:1531–1542
Yu H-K (2005) Weighted fuzzy time series models for TAIEX forecasting. Physica A 349:609–624
Zadeh LA (1965) Fuzzy sets. Inf Control 8:338–353. https://doi.org/10.1016/S0019-9958(65)90241-X
Zeng S, Chen S-M, Teng MO (2019) Fuzzy forecasting based on linear combinations of independent variables, subtractive clustering algorithm and artificial bee colony algorithm. Inf Sci 484:350–366

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.