JTAM (Jurnal Teori dan Aplikasi Matematika)
http://journal.ummat.ac.id/index.php/jtam
p-ISSN 2597-7512 | e-ISSN 2614-1175
Vol. 8, No. 3, July 2024, pp. 754-766

Dual Optimization of Weighted Fuzzy Time-Series Forecasting: Particle Swarm Optimization and Lagrange Quadratic Programming

Armando Jacquis Federal Zamelina[1], Suci Astutik[1], Rahma Fitriani[1]
[1]Department of Statistics, Brawijaya University, Indonesia
sucisp@ub.ac.id

Article History:
Received : 19-03-2024
Revised : 06-05-2024
Accepted : 07-05-2024
Online : 17-07-2024

Keywords:
Forecasting;
Weighted Fuzzy Time Series;
Particle Swarm Optimization;
Lagrange Quadratic.

ABSTRACT

Time series Forecasting is one of crucial techniques that helps with strategic decision-making and mitigating potential risks –One of which is Weighted fuzzy time series (WFTS). Moreover, the interval length of the WFTS plays a crucial role in its modelization and accuracy in predicting future values. Therefore, this research implements a dual optimization on WFTS, which are (1) Particle Swarm Optimization to find the optimum interval length of the WFTS and (2) a Lagrange quadratic to optimize the weight of the fuzzy interval. In this research, a univariate Average Air Temperature located in Malang is used to perform forecasting model. The dataset is taken from BMKG-Indonesia. This research aims to acquire an optimized interval length on fuzzy time series forecasting, i.e., improving its accuracy by finding the optimal interval length. Based on the result, the proposed dual optimization model outperforms the classical WFTS on forecasting. The proposed model excels based on the evaluation matrix values. It has been noticed also that implementing PSO to find the optimum interval length has improved the accuracy of the classical WFTS. The classical WFTS has MAPE and RMSE of 2.4 and 0.73, respectively, while the proposed dual optimized model has 1.01 and 0.3. This approach identifies the best interval values and provides optimum weights related to each data point, providing solid insights for air temperature forecasting.

Crossref
https://doi.org/10.31764/jtam.v8i3.22554

—————————— ◆ ——————————

A. INTRODUCTION

The field of system modeling encompasses a significant and broad range of research areas, one of which is time series forecasting. The fundamental objective of time series forecasting is to extract underlying patterns from data and make predictions about future values using past observations (Liu et al. 2021). Time series forecasting is widely utilized in diverse industries such as weather forecasting, climate forecasting, healthcare forecasting, finance forecasting, social studies forecasting, and others (Kumar Jha & Pande 2021; Lim & Zohren 2021; Zamelina et al., 2022). The previous research describes the data for time series analysis techniques and modeling techniques in order to forecast and provide insights for strategic decision-making. Those approaches enable decision-makers to mitigate potential risks and arrive at more advantageous ones.

There are several methods to perform forecasting. The Autoregressive Integrated Moving Average (ARIMA) model is frequently utilized in linear time series forecasting techniques and is applied in several fields such as business, agriculture, social sciences, etc. (Mishra et al. 2021;

Novianti et al., 2022). ARIMA models have been found to provide high accuracy levels when forecasting time series data with relative stationarity. The Seasonal ARIMA (SARIMA) model is used when dealing with a univariate time series with trend and seasonal components (Sirisha et al., 2022). That study shows the RMSE value of ARIMA and SARIMA models, 8.68 and 7.27, respectively. However, those approaches assumed that the provided time series contains no missing data and has stationary. It is worth noting that numerous real-world time series data contain challenging nonlinear patterns that the ARIMA and SARIMA modeling may not successfully capture. Therefore, SARIMA models are powerful for capturing seasonal variations, but they require careful parameter tuning. Researchers often choose between ARIMA and SARIMA based on the presence of seasonality in their data. In this research the Air Temperature data has seasonality, besides, it needs appropriate selection of seasonal differencing, seasonal autoregressive and seasonal moving average values for SARIMA to be implemented.

In regard of the cautious SARIMA parameter tuning and data characteristics limitations, a machine learning approaches have emerged as a viable technique for addressing time series approximation challenges. These techniques leverage historical time series data to predict the value of future data points (Song & Chissom 1993). The topic of the Fuzzy Times series has garnered significant attention in contemporary research. Additionally, based on the research of Alpaslan et al. (2012) while comparing the seasonal FTS and the SARIMA to time series of the amount of sulfur dioxide in Ankara province, the RMSE of seasonal FTS and SARIMA are 2.88 and 9.62 respectively. Despite the demonstrated effectiveness of fuzzy time series in diverse applications, it is essential to acknowledge its inherent limits. A primary drawback of the conventional fuzzy time series is in its treatment of prior data points as equal entities.

Rozy et al. (2023) have recently introduced a novel approach to address the lack of historical trend consideration in the procedure for defuzzification of the WFTS model. A novel approach was introduced by the researchers to enhance the WFTS model by the incorporation of the Lagrange Quadratic Programming (LQP) optimization technique for weight estimation. Their study displays a relatively small value of MAPE 0.61%. That study uses Lagrange Quadratic techniques in the context of WFTS to determine the optimal solution for the objective function in order to minimize the model's error. Mathematical modeling of WFTS significant improvements. On the other hand, Chen & Chen (2015) and Tinh et al. (2021) state that the interval length of the fuzzy set impacts the efficacy of the Fuzzy Time Series model in forecasting. Besides, Rozy et al. (2023) are still using static or equal-length intervals.

The process of setting parameter values of interval length through trial and error is not practicable because there are so many probable combinations of values and even infinite values. As a result, an optimization technique is required to find the best parameter values that do not necessitate too many experiments and take a relatively quick time toward the optimum results. Metaheuristics is one of the techniques frequently used to solve the problem of identifying the optimal global solution (Hussain et al. 2019). Particle Swarm Optimization (PSO) is one of the metaheuristic methods that can work efficiently in identifying optimum interval length (Surono et al. 2022; Tinh et al. 2021). Ariyanto et al. (2021) found that the PSO employs to set non-static length of intervals, leading to a lower RMSE value than the conventional forecasting methods.

Moreover, the phenomenon of global warming has recently received significant interest from the scientific community due to its observed correlation with the increase in atmospheric temperatures. The prediction of Air Temperature holds significant importance within the field of weather forecasting, as it serves as an essential component in preserving human lives and protecting valuable properties. Increases or decreases in air temperature of significant magnitude have the potential to induce harmful impacts on plant and animal life. Precisely predicting atmospheric temperature is crucial, given its substantial impact on various sectors (Kang et al., 2019; Musashi et al, 2018) it affects public safety, healthcare, agriculture, energy management, transportation, environmental conservation, infrastructure planning, tourism, the economy, and more.

Therefore, there are two main approaches to optimize the WFTS, one of which are interval length and weights for defuzzification steps. The Particle Swarm Optimization (PSO) enhance the approach by finding the optimum interval length and the Lagrange Quadratic Programming (LQP) address the weight optimizations by detecting optimum points resulting from changes in patterns in the time series data. Additionally, having an accurate Air Temperature forecasting helps decision-makers to take decision beforehand of the future temperature change. The used Air temperature data is taken from the online Database-BMKG, Indonesia, over three (03) years. It is anticipated that the outcomes of this study will afterward serve to forecast the impact of Air temperature-related events in Malang.

B. METHODS

Forecasting is a methodological approach that predicts future events or outcomes based on historical data and relevant information (Cerqueira, Torgo, and Mozetič 2020). Based on Lim and Zohren (2021), time-series forecasting models serve to make predictions about future values of a target variable, denoted as yt, at a particular time t. In its most basic form, one-step-ahead forecasting models can be represented as:

$$y_{t+1} = f(y_t - k) \tag{1}$$

where y_{t+1} is the model forecast, y_t - k is the observations of the targets over a look-back window k, and $f(\ldots)$ is the forecast function learned by the model.

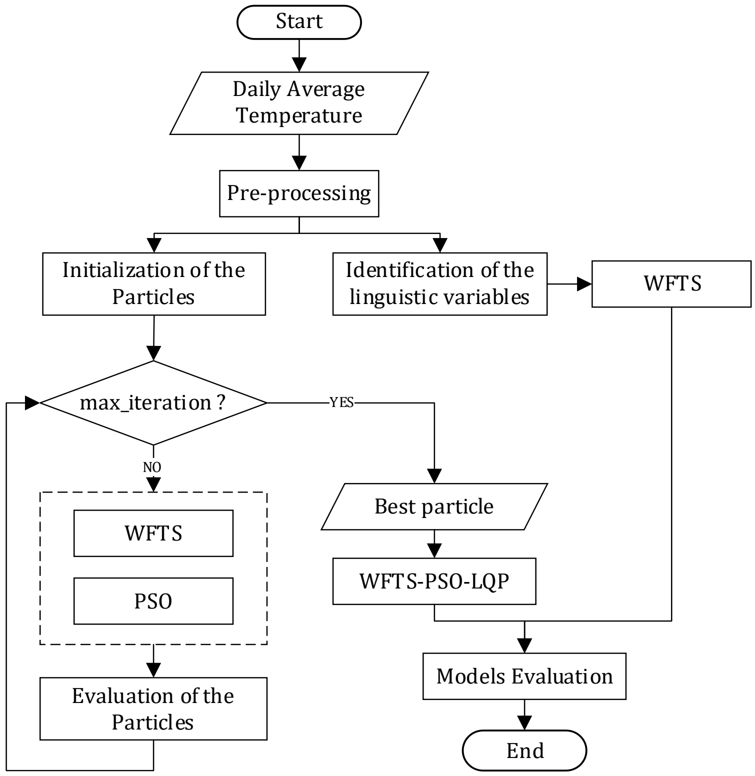

Figure 1. Research flow diagram

Figure 1 illustrates the project's workflow. There are three major steps: First, Pre-processing data, second, determining the linguistic variables of the WFTS and the proposed model, and last models Evaluations the pre-processing steps involve handling the missing values and the outliers. About the second step, the proposed model starts by initializing the initial particles, where each particle has different interval length or different linguistic variables values. Next, identifying the best particle based on MAPE value over PSO iteration on each particle and followed by the implementation of the LQP. Besides, as the classical WFTS uses equal-length interval, its step is directly setting the linguistic variables based on the interval values. Lastly, the third step is by comparing the two models –The classical WFTS and the proposed model WFTS-PSO-LQP.

1. Dataset and Pre-processing data

The average daily temperature data in Celsius is extracted from the online database, BMKG (Badan Meteorologi, Klimatologi dan Geofisika), located in Malang, Indonesia. The dataset ranges from July 1, 2020, to June 1, 2023, with 1066 rows of Average temperature data during almost three years. We convert the raw daily AT data into a more practical format for forecasting. One of the initial pre-processing steps is to address missing values. For this study, we employed the 'rolling mean' scheme with a window size of 7 to address the missing data. That means we calculate the mean of 7 days both preceding and following the missing values. Next, the dataset will be partitioned into training and testing sets.

2. The Weighted Fuzzy Time Series

Song & Chissom (1993) were the first to develop the concept of fuzzy time series. Fuzzy Time Series (FTS) is a methodology employed in the field of forecasting and decision-making, in which historical data and fuzzy functions are combined to make predictions about future values. This fuzziness arises from the inherent ambiguity present in the datasets. The Weighted Fuzzy Time Series (WFTS) technique is a development of FTS method (Yu 2005). WFTS adds weight to each fuzzy relationship, so it emphasizes the varied significance of the sequence of fuzzy relations. The steps for forecasting by using Chen's algorithm can be described into the following steps (Suhartono et al., 2011):

Step 1. Defining the Universe of Discourse U based on the maximum value (D_{min}) and minimum value (D_{max}) from the data set and two positive numbers, d1 and d2.

$$U = [D_{min} - d_1,\ D_{min} - d_2] \tag{2}$$

Step 2. Partitioning the Universe of Discourse. Dividing the universal set into several subsets with the same range size and then forming a fuzzy set to the subsets $u_1, u_2, \ldots, u_n$.

$$sturges = 1 + 3.322 \log n \tag{3}$$

Step 3. Defining the fuzzy sets Ai, which are established based on the subsets of the universe of discourse. Fuzzification is the process of converting the crisp (exact) value into fuzzy sets. The fuzzification obtained from one set is shown by the Equation (4):

$$\begin{gathered} A_1 = \frac{a_{11}}{u_1} + \frac{a_{12}}{u_2} + \ldots + \frac{a_{1n}}{u_n} \\ A_1 = \frac{a_{21}}{u_1} + \frac{a_{22}}{u_2} + \ldots + \frac{a_{2n}}{u_n}, \\ \ldots \\ A_k = \frac{a_{k1}}{u_1} + \frac{a_{k2}}{u_2} + \ldots + \frac{a_{kn}}{u_n} \end{gathered} \tag{4}$$

with A_i is the membership degree value, where A_{ij} j the members$i \leqslant k$, and $1 \leqslant j \leqslant n$. For instance, a data point is involved in a fuzzy set A_j when its degree of membership in A_j is maximized.

Step 4. Developing Fuzzy Logical Relationships Group. The grouping is determined by the present states of the data related to its next state fuzzy logical relationships. For example, $A_i \rightarrow A_b$, $A_i \rightarrow A_d$, $A_i \rightarrow A_e$ then $A_i \rightarrow A_b, A_d, A_e$.

Step 5. Calculating the forecasted values. Consider $(t - 1) = A_i$, then The value of $F(t)$ is calculated by considering the following scenarios:

1) Case 1: The fuzzy logical sequence consists of only a single fuzzy logical relationship. If $A_i \rightarrow A_j$, then the prediction value (t) can be considered equal to A_j.

2) Case 2: The series of fuzzy logical comprises many fuzzy logical relationships. If $A_i \rightarrow A_i$, A_j,..., A_k, then the prediction value $F(t)$ is taken from $(n_1 + n_2 + \cdots + n_p)/p$, where $n_1, n_2, \dots, n_i$ are the midpoints of the intervals $u_1, u_2, \dots, u_i$, respectively.
3) Case 3: If the fuzzy logical relationship sequence does not exist, then the forecasted value remains the same with the current logical relationships.

Step 6. Performing defuzzification. It is the reverse process of the fuzzification. Consider that the forecast of $F(t)$ are $A_{j1}, A_{j2}, \dots, A_{jk}$. The defuzzified result is the same as the midpoint value matrix of $A_{j1}, A_{j2}, \dots, A_{jk}$:

$$M(t) = [mj1, mj2, \dots, mjk] \tag{5}$$

Step 7. Assigning weights (Yu 2005).

$$w(t) = [w_1^{'}, w_2^{'}, \dots, w_k^{'}] = \left[\frac{w_1}{\sum_{h=1}^{k} w_h}, \frac{w_2}{\sum_{h=1}^{k} w_h}, \dots, \frac{w_k}{\sum_{h=1}^{k} w_h}\right] \tag{6}$$

Step 8. Finding predicted values. To compute the prediction value, the weighted model is obtained by multiplying the defuzzified matrix with the transpose of the weight matrix. According to Surono et al. (2022), a differencing technique applies to compare the observed data with the midpoint values generated inside each interval class for the purpose of predicting:

$$\hat{F}(t) = M(t) \times w(t)^T, \text{ and} \tag{7}$$

$$\hat{F}(t+1) = F(t+1) \pm |diff(X(t), m_i) \tag{8}$$

where $m_1, m_2, \dots, m_n$ is the middle value of the interval.

3. Lagrange Quadratic Weighted Fuzzy Time Series

The Lagrange Quadratic optimizes the weight of the Weighted Fuzzy Time Series model. Lagrange Quadratic Programming in WFTS (WFTS-LQP) is set from a combination of two approaches, namely Lagrange Multiplier and Quadratic Programming (Rozy et al., 2023). Lagrange Multiplier was first introduced by Joseph Louis Lagrange (1736-1813), which was used to optimize real-valued functions. That means a method of evaluating maximum or minimum function problems which are formed from changing a constrained extreme point problem into a constraint free extreme problem. To implement the Lagrange Quadratic weight on FTS, steps 1-4 remain the same with the FTS, and the difference is on calculating the weight before defuzzification.

Step 1 – 7. Same with the Weighted Fuzzy time series processes.

Step 8. Identifying the optimal point (represented as $x^* \equiv (x, y)$) inside a multidimensional space that achieves local optimizations of the merit function *f(x)* while adhering to the restriction *g(x)=0*. Determining the stationary point in this constrained optimization problem is done by modeling it into a Lagrange function, which can be seen on Equation (9):

$$F(y) = f(y) + \sum_{i=1}^{m} \lambda_i g_i(y) \quad (9)$$

where $\lambda_1, \lambda_2, \dots, \lambda_m$ is the Lagrange multiplier.

Step 9. Defining the objective function and constraints for the weighted fuzzy time series using Langrage Quadratic:

$$L(\omega_{i,j}, \lambda_j) = \sum_{i=1}^{n} \omega_{i,j}^2 u_{i,j} + 2\lambda_i \left(\sum_{i=1}^{n} \omega_{i,j} - 1\right) \quad (10)$$

with *j = 1,2, 3,…, n*.

Step 10. Same with the step 8 of the conventional Weighted Fuzzy Time Series algorithm.

4. Particle Swarm Optimization

James Kennedy and Russell Ebenhart introduced the Particle Swarm Optimization (PSO) algorithm in 1995. It is one of many metaheuristic algorithms that can be applied to address optimization problems (Rocha, 2021). The researcher added that it is an optimization technique that derives inspiration from the collective behaviour observed in flocks of birds and schools of fish as they navigate and find solutions to complex nonlinear problems.

To find the optimum result, the particle (birds) undergoes two distinct forms of learning. Each particle will acquire knowledge through its own movement as well as from the collective experiences of other particles. Cognitive learning yields the outcome whereby particles retain the memory of the optimal solution (Particle best), represented as P_{best}. In the context of social learning, the optimal outcome (Global best) is represented as G_{best}. The following formula can be used to compute the movement of particles into a new position:

$$x_{i,j}^{t+1} = x_{i,j}^{t} + v_{i,j}^{t+1} \quad (11)$$

where $x_{i,j}^{t}$ is the actual position of the particle *i* and dimension *j* in the iteration *t*. And $v_{i,j}^{t}$ represnts the velocity of particle *i* of dimension *j* in the iteration *t*. The velocity and the position will always be updated. Equation (12) is used to update the velocity.

$$v_{i,j}^{t+1} = \omega . v_{i,j}^{t} + c_1 . r_1 \left(P_{best}{}_{i,j}^{t} - x_{i,j}^{t} \right) + c_2 . r_2 (G_{best}{}_{g}^{t} - x_{i,j}^{t}) \quad (12)$$

where ω is the inertia weight, c_1 represents the self-cognition, c_2 represents the social cognition (acceleration coefficient), r_1 and r_2 are a random value between 0 and 1. $\mathrm{P}_{\mathrm{best}}{}_{\mathrm{i,j}}^{\mathrm{t}}$ and $G_{best}{}_{g}^{t}$ denotes the best personal and best global of particle *i* and dimension *j* in the iteration *t*, respectively. $x_{i,j}^{t}$ represent the actual position of the particle *i* of dimension *j* at iteration *t*. The process phase regarding Particle Swarm Optimization is illustrated in Figure 1 (Qiu, Zhang, and Ping 2015). The process of the PSO starts with the initialization of the particles. Each particle is then integrated to a WFTS model, where its MAPE value is the fitness value of a particle. The particle's values (The position of a particle) are then update based on P_{best}, G_{best}, and velocity. The scenario is repeated until the end of the iteration or the stop criteria is met.

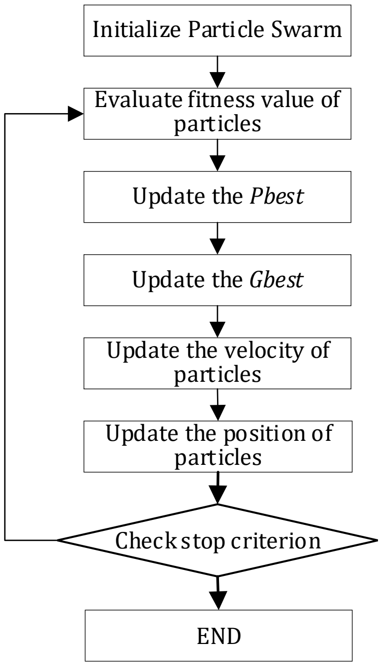

Figure 2. The search procedure of the PSO.

5. Performance Evaluation

Once the models have been built, we assess and compare the predicted results using evaluation measures. Two error metrics were employed in this study to assess the models. The two metrics are Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). Equations (13) and (14) represent the formulas for calculating the error metrics MAPE and RMSE, respectively (Rocha 2021).

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{x_t - \hat{x}_t}{x_t}\right| \times 100\% \tag{13}$$

$$RMSE = \sqrt{\left(\frac{1}{n}\right)\sum_{i=1}^{n}(x_t - \hat{x}_t)^2} \tag{14}$$

where n is the amount of data, x_t is the actual value at time t, and $\hat{x}_t$ is the predicted value at time t.

C. RESULT AND DISCUSSION

This section describes the outputs and discusses the model's performance. This study utilizes Chen's Fuzzy Time Series model basis to forecast average daily temperatures. The model Chen's fuzzy model is optimized using weight, PSO, and LQP. We utilize qualitative and quantitative frameworks to achieve optimal results for the models.

1. Experiment Model Training

The research is conducting a univariate time series analysis. There is only one feature of the AT time series. The model development procedure begins by undergoing pre-processing and then establishing the initial hyperparameters to achieve the most effective model for forecasting. The two models have distinct parameters because PSO is utilized in the proposed model to determine the optimal interval length of the optimized WFTS. The first step for both models is setting the Universe of discourse. The following are the distinct steps for the Chen's conventional WFTS and the optimized WFTS:

a. The classical Weighted Fuzzy Time Series

Chen's conventional fuzzy time series model uses the same interval length for all fuzzy sets. Its interval length is obtained from the initial value of the discourse plus the final value of the discourse divided by the result of the dimension particle from (3). The number of linguistic variables for the classical WFTS within the discourse range is then equal to 9. By performing the weighted fuzzy time series procedure, which is described in section B.2, we got the weight and the forecasted values from the WFTS model.

b. The proposed model

The determination of the number of linguistic variables is the same as the conventional fuzzy time series, but the interval fuzzy sets do not have an equal-length intervals. The initial interval values for the proposed model are obtained from a uniform randomization within the range of the Universe of Discourse. The following are the hyperparameters for the PSO, which is gotten from a random search:

1) Number of particles: 5
2) Number of iterations: 20
3) c_1: 1
4) c_2: 1.5
5) w: 0.3
6) r_1 and r_2: Random variables that change through iteration.

Those hyperparameter values produce an optimal model (neither underfitting nor overfitting). As the number of particles is five (5), thus we have 5 different sets of linguistic variables between each particle, i.e., each particle has A_1-A_9 with different values of fuzzy sets length. From that, we perform the step of the Lagrange Quadratic Weighted Fuzzy Time Series by using one particle that has the lowest MAPE value. The number of functions for Lagrange quadratic, described in Equation (10), is the same as the number of the linguistic variables.

2. Identification of the linguistic variables

Firstly, based on (2), we got the value of the Universe of Discourse for all models U = [19.1, 27.9]. As the conventional fuzzy time series uses the equal length of the interval, thus the possible interval length within the universe of discourse range is 1. Therefore, the number of linguistic variables for the classical WFTS is 9: A_1= [19, 20), A_2= [20, 21), A_3= [21, 22), A_4= [22, 23), A_5= [23, 24), A_6= [24, 25), A_7= [25, 26), A_8= [26, 27), A_9= [27, 28].

For the optimized model, each particle has different values of fuzzy set length. The following is an example of the linguistic variable of Particle number 4 before performing the optimization with PSO: A_1= [19.1, 19.50), A_2= [19.50, 20.6), A_3= [20.6, 20.85), A_4= [20.85, 23.11), A_5= [23.11, 23.62), A_6= [23.62, 24.31), A_7= [24.31, 24.44), A_8= [24.44, 26), A_9= [26, 27.9]. After the last iteration, the following are the fuzzy sets values of the Particle 4: A_1= [19.1, 19.52), A_2= [19.52, 20.58), A_3= [20.58, 20.86), A_4= [20.86, 23.19), A_5= [23.19, 23.59), A_6= [23.59, 24.31), A_7= [24.31, 24.5), A_8= [24.5, 26.04), A_9= [26.04, 27.9].

3. Selection of the best particle with PSO

There are five particles, and each particle is implemented into one running of WFTS model in iterative way. They are also implemented inside the PSO function. Thus, after each PSO iteration, we get the performance evaluation of each particle. MAPE is the matrix used to evaluate each particle's efficiency. Table 1 displays the evaluation of each particle from the initial iteration until the last iteration. We noticed that the error values decreased between the initial and final iterations. Initially, the lowest MAPE is 2,4387, and the highest is 3.1767. Upon the last iteration, all particles tend towards the least error values. All particles tend to move towards the best position because of the social component of the PSO –see (12). Particle 4 has the lowest error value at final iteration. Its MAPE value is equal to 2.3013. It was noticed that the PSO algorithm improved forecasting accuracy during the fuzzy length optimization process. The particle 4 is then used as the fuzzy interval set for the WFTS with Lagrange Quadratic, as shown in Table 1.

Table 1. MAPE Calculation Based on the Model Iteration in Training Set

Particles	**MAPE**	
	Initial iteration	**Final iteration**
Particle 1	2.5633	2.3175
Particle 2	3.1767	2.3135
Particle 3	2.7680	2.3014
Particle 4	2.4387	2.3013
Particle 5	2.6037	2.3077

4. Estimation of the weight through LQP

The number of Lagrange quadratic equations is the same as the number of fuzzy sets that have previously been calculated. Based on (10), the following is an example of one equation for the interval A_1:

$$L(\omega_{i,1}, \lambda_1) = \sum_{i=1}^{9} \omega_{i,1}^2 \mu_{i,1} + 2\lambda_1 \left(\sum_{i=1}^{9} \omega_{i,1} - 1\right) \quad (15)$$

to get the solution, we perform a partial derivative, on each Lagrange quadratic equation, with respect to ω and λ. The following is the example of the class interval A_1:

$$\frac{\partial L(\omega_{i,1}, \lambda_1)}{\partial \omega_i} = 2 \sum_{i=1}^{9} 2\omega_{i,1} \mu_{i,1} + 2\lambda_1 = 0 \quad (16)$$

$$\frac{\partial L(\omega_{i,1},\lambda_1)}{\partial \lambda_i} = \sum_{i=1}^{9} 2\omega_{i,1} - 2 = 0 \tag{17}$$

in the final stage, the weights and membership values for each interval class are multiplied according to step 10 of the Lagrange Quadratic Weighted Fuzzy time series.

5. Comparison of the Proposed model with the conventional Weighted Fuzzy Time Series

Table 2 shows the assessment result of the model on the test data. The forecasting method used is a recursive forecasting over 30 days. Lower MAPE and RMSE values reflect an excellent model performance. The MAPE for WFTS without PSO is 2.4007, implying that, on average, its forecasts differ by 2.4% from the actual values. And its RMSE is 0.7323. After implementing the PSO on the WFTS, the MAPE and the RMSE have reduced to 2.3101 and 0.7311, respectively. Furthermore, the MAPE of the proposed dual optimization of the WFTS is around 1.05% and its RMSE is roughly 0.30. Thus, WFTS-PSO-LQP exceeds the traditional WFTS in both MAPE and RMSE metrics, as shown in Table 2.

Table 2. Evaluation of the Model's Forecasting

Model	MAPE	RMSE
WFTS	2.4007	0.7323
WFTS-PSO	2.3101	0.7311
WFTS-PSO-LQP	1.0591	0.3002

Figure 3 illustrates a qualitative performance plot comparing the forecasting of 30 days using the classical WFTS and the new optimized models. The data visualization in Figure 3 shows a strong correspondence between the plot lines of Actual AT and the proposed model. The proposed dual optimization surpasses the classical model.

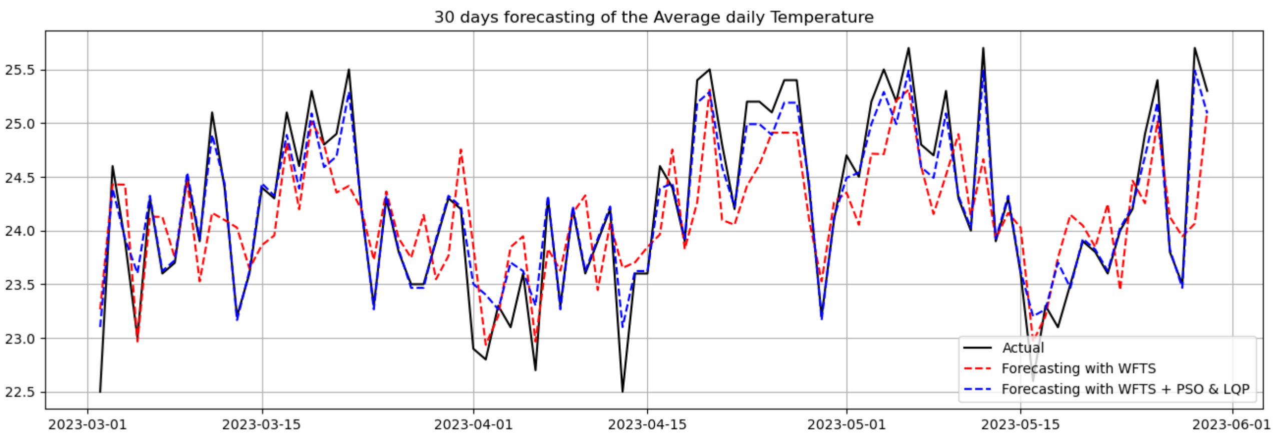

Figure. 3. Plotting test set of the actual AT data related to the WFTS and the Proposed model.

D. CONCLUSION AND SUGGESTIONS

Forecasting an accurate Air Temperature is essential to human life as it affects our daily activities in various sectors. This research investigates an approach that implement dual optimization for Air temperature forecasting with a basis Weighted Fuzzy Time Series (WFTS) technique. The first optimization is by enhancing the fuzzy set interval values, and the second is through the weight of the linguistic variables. Finding the optimum interval length of the WFTS using PSO has improved the accuracy of the fuzzy model where the MAPE value of the forecast Air Temperature of the classical WFTS and the WFTS-PSO are 2.4007 and 2.3101, respectively. Moreover, the weight of the linguistic variables is optimized through LQP. The evaluations of the classical WFTS are MAPE= 2.4007, RMSE= 0.7323, and the proposed model are MAPE=1.0591, RMSE= 0.3002. Based on the result, it is noticed that the proposed dual optimization model excels the classical Weighted Fuzzy Time Series when applied to the test set of 30 days ahead. This approach determines the best interval values and provides ideal weights correlated to each data point, which provides strong insights for air temperature forecasting and assisting decision-makers in making decisions ahead of time about future temperature change. As the optimization of the fuzzy time series might be conducted through the interval length and the order or the lookback value for the FLRG, and this paper only covers the interval optimization, thus future research may conduct further study on optimizing the lookback value. In addition, implementing LQP on multivariate dataset still a challenge needs to be studied further.

REFERENCES

Alpaslan, Faruk, Ozge Cagcag, C. H. Aladag, U. Yolcu, E. Egrioglu, F Alpaslan, and O Cagcag. 2012. A Novel Seasonal Fuzzy Time Series Method. Vol. 41. https://zbmath.org/1278.62137

Ariyanto, R., R. H. Tjahjana, and T. Udjiani. 2021. "Forecasting Retail Sales Based on Cheng Fuzzy Time Series and Particle Swarm Optimization Clustering Algorithm." in *Journal of Physics: Conference Series*. Vol. 1918. IOP Publishing Ltd. doi: 10.1088/1742-6596/1918/4/042032

Cerqueira, Vitor, Luis Torgo, and Igor Mozetič. 2020. "Evaluating Time Series Forecasting Models: An Empirical Study on Performance Estimation Methods." *Machine Learning* 109(11):1997–2028. doi: 10.1007/s10994-020-05910-7.

Chen, Mu Yen, and Bo Tsuen Chen. 2015. "A Hybrid Fuzzy Time Series Model Based on Granular Computing for Stock Price Forecasting." *Information Sciences* 294:227–41. doi: 10.1016/j.ins.2014.09.038.

Hussain, Kashif, Mohd Najib Mohd Salleh, Shi Cheng, and Yuhui Shi. 2019. "Metaheuristic Research: A Comprehensive Survey." *Artificial Intelligence Review* 52(4):2191–2233. doi: 10.1007/s10462-017-9605-z.

Kang, Fei, Junjie Li, and Jianghong Dai. 2019. "Prediction of Long-Term Temperature Effect in Structural Health Monitoring of Concrete Dams Using Support Vector Machines with Jaya Optimizer and Salp Swarm Algorithms." *Advances in Engineering Software* 131:60–76. doi: 10.1016/j.advengsoft.2019.03.003.

Kumar Jha, Bineet, and Shilpa Pande. 2021. "Time Series Forecasting Model for Supermarket Sales Using FB-Prophet." Pp. 547–54 in *Proceedings - 5th International Conference on Computing Methodologies and Communication, ICCMC 2021*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICCMC51019.2021.9418033

Lim, Bryan, and Stefan Zohren. 2021. "Time-Series Forecasting with Deep Learning: A Survey." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 379(2194). https://doi.org/10.1098/rsta.2020.0209

Liu, Zhenyu, Zhengtong Zhu, Jing Gao, and Cheng Xu. 2021. "Forecast Methods for Time Series Data: A Survey." *IEEE Access* 9:91896–912. doi: 10.1109/ACCESS.2021.3091162.

Mishra, Pradeep, A. M. G. Al Khatib, Iqra Sardar, Jamal Mohammed, Kadir Karakaya, Abhiram Dash, Monika Ray, Lakshmi Narsimhaiah, and Anurag Dubey. 2021. "Modeling and Forecasting of Sugarcane Production in India." *Sugar Tech* 23(6):1317–24. doi: 10.1007/s12355-021-01004-3.

Musashi, Jaka Pratama, Henny Pramoedyo, and Rahma Fitriani. 2018. "Comparison of Inverse Distance Weighted and Natural Neighbor Interpolation Method at Air Temperature Data in Malang Region." *CAUCHY: Jurnal Matematika Murni Dan Aplikasi* 5(2):48–54. doi: 10.18860/ca.v5i2.4722.

Novianti, Trisita, Issa Dyah Utami, and Wike Agustin Prima Dania. 2022. "Forecasting of Salt Demand Using ARIMA Model to Prevent the Bullwhip Effect in Salt Supply Chain." in *Journal of Physics: Conference Series*. Vol. 2193. Institute of Physics. doi: 10.1088/1742-6596/2193/1/012035

Qiu, Wangren, Chunhua Zhang, and Zhang Ping. 2015. "Generalized Fuzzy Time Series Forecasting Model Enhanced with Particle Swarm Optimization." *International Journal of U- and e-Service, Science and Technology* 8(5):129–40. doi: 10.14257/ijunesst.2015.8.5.11.

Rocha, Lidia. 2021. *Comparison between Meta-Heuristic Algorithms for Path Planning*. https://doi.org/10.5753/wtdr_ctdr.2020.14950

Rozy, Fachrur Agus, Ni Wayan Surya Wardhani, and Solimun. 2023. "Development of Accuracy for the Weighted Fuzzy Time Series Forecasting Model Using Lagrange Quadratic Programming." *JTAM (Jurnal Teori Dan Aplikasi Matematika)* 7(4):1193–1203. doi: 10.31764/jtam.v7i4.16783.

Sirisha, Uppala Meena, Manjula C. Belavagi, and Girija Attigeri. 2022. "Profit Prediction Using ARIMA, SARIMA and LSTM Models in Time Series Forecasting: A Comparison." *IEEE Access* 10:124715–27. doi: 10.1109/ACCESS.2022.3224938.

Song, Qiang, and Brad S. Chissom. 1993. "Fuzzy Time Series and Its Models." *Fuzzy Sets and Systems* 54(3):269–77. doi: 10.1016/0165-0114(93)90372-O.

Suhartono, Muhammad Hisyam Lee, and Hossein Javedani. 2011. *A Weighted Fuzzy Integrated Time Series for Forecasting Tourist Arrivals*. Vol. 252. https://doi.org/10.1007/978-3-642-25453-6_19

Surono, Sugiyarto, Khang Wen Goh, Choo Wou Onn, Afif Nurraihan, Nauval Satriani Siregar, A. Borumand Saeid, and Tommy Tanu Wijaya. 2022. "Optimization of Markov Weighted Fuzzy Time Series Forecasting Using Genetic Algorithm (GA) and Particle Swarm Optimization (PSO)." *Emerging Science Journal* 6(6):1375–93. doi: 10.28991/ESJ-2022-06-06-010.

Tinh, Nghiem Van, Nguyen Cong Dieu, Nguyen Tien Duy, and Tran Thi Thanh. 2021. "Improved Fuzzy Time Series Forecasting Model Based on Optimal Lengths of Intervals Using Hedge Algebras and Particle Swarm Optimization." *Advances in Science, Technology and Engineering Systems Journal* 6(1):1286–97. doi: 10.25046/aj0601147.

Wang, Shixiong, Chongshou Li, and Andrew Lim. 2019. "Why Are the ARIMA and SARIMA Not Sufficient." https://doi.org/10.48550/arXiv.1904.07632

Yu, Hui Kuang. 2005. "Weighted Fuzzy Time Series Models for TAIEX Forecasting." *Physica A: Statistical Mechanics and Its Applications* 349(3–4):609–24. doi: 10.1016/j.physa.2004.11.006.

Zamelina, Armando Jacquis Federal, Didit Adytia, and Annas Wahyu Ramadhan. 2022. "Forecasting of Maximum Temperature by Using ANFIS and GRU Algorithms: Case Study in Jakarta, Indonesia." Pp. 222–27 in *2022 10th International Conference on Information and Communication Technology, ICoICT 2022*. Institute of Electrical and Electronics Engineers Inc. doi: https://doi.org/10.1109/ICoICT55009.2022.9914885