IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 24, Issue 1, Ser. III (Jan. –Feb. 2022), PP 31-41
www.iosrjournals.org

Fusion of Soft Computing and Chaos Theory to Build A Model For Prediction Quality Of Wastewater In Industrial Zones

Dung Vuong Quoc*, Loi Nguyen Tien, Nguyet Do Thi Minh, Thuy Tran Thi, Minh Nguyen Quang, Binh Nguyen Thi Thanh

Center of Information Technology, Hanoi university of Industry, Number 298, Cau Dien Street, Bac Tu Liem Distric, Ha Noi, Vietnam

Abstract: *In this paper, we present a novel method for detecting environmental pollution of waste-water in industrial zones. Firstly, the quality of waste-water data is filtered by an adaptive filter. After that, the false nearest neighbor and average mutual information algorithms are applied to find embedding dimension space and time delay of waste-water quality time series to form training and testing set for the model. Finally, the Support Vector Regression and Fuzzy logic is implemented to build model for prediction quality of waste-water in industrial zones. Four main parameters at the waste-water processing station of Nittoku paper factory in the Kim Bang district, Ha Nam province, Vietnam have been used to test proposed method. The experimental results show that the proposed model is high accuracy and short training time, to helps waste-water processing station operators take early action and avoid environmental pollution.*

Keywords: *Support Vector Regression, Fuzzy logic, pollution, waste water, time delay, embedding dimension space.*

Date of Submission: 14-02-2022 Date of Acceptance: 28-02-2022

I. Introduction

According to Circular 24 of the Vietnamese Ministry of Natural Resources and Environment, all industrial zones that is discharged waste into the environment with a flow rate is greater than *1000 m*^{3} per day must be installed an automatic monitoring system for waste-water quality. At the waste-water processing station of an industrial zone, the operator based on certain measurement and monitoring values, must take proper action when any criterion of waste-water is over the permitted threshold value. Therefore, waste-water that does not satisfy this standard that has already discharged to environment lead to environment pollution. Hence, to avoid the environmental pollution, we can build a model based on historical data collected by data-loggers to predict future quality of the waste water; the model could help operators of the waste water processing stations take early action to make sure that the quality of the waste water always satisfies the standard requirements.

Predicting future time series values from the past is applied in many fields, such as the following: economics where forecasting stock price helps investors choose the best time to invest in the stock market to get the highest profit; predicting exchange rates which helps import-export businesses choose appropriate times to import or export products (Wang, Y. F., 2002; Castillo, Oscar Melin, Patricia, 2002); energy, where predicting the wind speed of a wind farm and the electrical load helps energy policymakers give a best plan to meet the requirement of clients (Lanzhen, L., Tan, S. et al., 2010; Hua, X., Zhang, D., Leung, S.C., st al., 2010), and so on.

Due to the importance of predictive work, scientists have successfully used a number of techniques in recent years to build a data series predictive model. In (Hua, X., Zhang, D., Leung, S.C, 2010) authors combined Kernel Regression (KR) and Function Link Artificial Neural Network (FLANN) to predict exchange rates from USD to GBP, INR and JPY. KR played role in filtering and FLANN was a model for prediction. The authors in (Hanias, M.P., Curtis, P.G., 2008) used chaos theory and reconstructed state space for predicting the exchange rates between USD and EUR. Fan-Yong Liu (et al. 2010) used a hybrid Discrete Wavelet Transform (DWT) and Support Vector Regression (SVR) to predict the exchange rates between CNY and USD. Firstly, he used DWT to decompose the time series data to a different time scale and then he chose an appropriate kernel function for SVR and prediction corresponding with each time scale. Finally, he synthesized the prediction result from different predicted time scale results. The authors in (Iokibe, T., Murata, S., Koyama, M et al. 1995) used a local fuzzy reconstruction method to predict exchange rates between JPY, USD and CAN. The authors in (Božić, J., Vukotić, S., Babić, D. et al. 2011) used a successful wavelet transform to filter noise in the exchange rate time series before using it for training and prediction based on Multi-Layer Feed Forward Neural Network. Weiping Liu (2018) used hybrid of neuron and fuzzy logic to predict exchange rates between JPY and USA.

Recently, scientific literature (Nghien Nguyen Ba and Ricardo Rodriguez Jorge et al. 2019) has been used SVR and Fuzzy logic to build model for prediction quality of waste-water in industrial zone. To specify number of the inputs for the model they used embedding dimension space. They used default time delay is one. The drawback of their proposed method is using original data time series that is contaminated by noise so the embedding dimension space of time series data is high and time delay is one so it is hard to distinguish two data points next to each other. This phenomenon caused not only the model' complexity but also loose accuracy prediction. Authors in [18] used SVR to build model to predict surface roughness in hole turning process of 3X13steel. They reported that SVR model is better accuracy prediction than response surface method (RSM). Authors in [19] used SVR for prediction the sorption capacity of lead (II) ions. They statement that SVR model is more accurate and generalized for prediction of the sorption capacity of lead (II) ions than multiple linear regression does. Authors in [20] compares the classical regression with SVR using some dataset base on root mean square error (RMSE) and R^2. They report that SVR achieves more accuracy prediction than classical model.

In this paper, the author proposes a fusion method to build model for prediction quality of waste-water in industrial zones. Firstly, he uses Hilbert – Huang transform as an adaptive filter to filter out noise from the original time series data. Next, the proposed method used false nearest neighbor and mutual information of cleaned time series data to find embedding dimension space and time delay. After that, forming input – output training pair from cleaned time series data and found embedding dimension space and time delay. Finally, method builds a model base on SVR and Fuzzy logic for prediction quality of the waste-water. Our proposed method not only decrease system's complexity but also increase accuracy of prediction.

II. Related Work

2.1. Finding the time delay

According to the literature review (Abarbanel, H.D., Brown, R., Sidorowich, J.J., Tsimring, L.S., 1993). If we select the time delay *T* too small, then two data points *s(n+jT)* and *s(n+(j+1)T)* will be so close to each other that we can not distinguish them from each other. Similarly, if we choose T so large, then *s(n+jT)* and *s(n+(j+1)T)* are completely independent of each other in a statistical sense. To determine proper time delay of a time series we can base on average mutual information. Assume we have two systems called *A* and *B*, and measured values from those system denoted by a_k, b_k. The mutual information between a_k and b_k is specified as equation (1) below:

$$I_{AB}(a_i, b_k) = log_2\left[\frac{P_{AB}(a_i,b_k)}{P_A(a_i)P_B(b_k)}\right] \qquad (1)$$

where, $P_A(a)$ is probability of observing *a* out of the set of all *A*, and the probability of finding *b* in a measurement of *B* is $P_B(b)$, and the joint probability of the measurement of *a* and *b* is $P_{AB}(a, b)$.

The average mutual information between measurements of any value a_i from a system *A*, and b_k from a system *B* is average over all possible measurements of $I_{AB}(a_i,b_k)$ and can be calculated by equation (2) below:

$$I_{AB}(T) = \sum_{a_i,b_k} P_{AB}(a_i, b_k) I_{AB}(a_i, b_k) \qquad (2)$$

To apply this definition into time series data *s(n)* which is measured from a physical system. We consider set of measurements *s(n)* as the set *A* and measurements a time lag *T*, *s(n+T)*, as the *B* set. The average mutual information between time series *s(n)* and *s(n+T)* can be evaluated as equation (3).

$$I(T) = \sum_{i=1}^{n} P[s(n), s(n+T)] log_2\left[\frac{P[s(n),s(n+T)]}{P[s(n)]P[s(n+T)]}\right] \qquad (3)$$

Hence, the average mutual information is a function of time lag *T* and *T* can be specified as the first min of the *I(T)*. If *I(T)* has not a minimum, then *T* will be chosen as 1.

2.2. Finding the embedding dimension space

The number of inputs for predictor is embedding dimension or can be thought as windows size (Frank, R.J., Davey, N., Hunt, S.P.). To valuate this parameter (Abarbanel, H.D., Brown, R., Sidorowich, J.J., Tsimring, L.S.) present the implementation of false nearest neighbors to find the minimum embed dimension. Assume we have a data time series, *s(n)*, the idea of the algorithm to combine sequence values of series together to form a set of vectors *V* of dimension *d (d=1, 2, 3, …)*. For example, vector *v(k)=[s(k), s(k+T), s(k+2T), …, s(k+(d-*

1)T], where *T* is time lag of time series. We use Euclidean distance to calculate the distance between two vectors. Formula (4) calculates the distance between vectors *v(k)* and *v(m)*.

$$D(d)_{km} = \sqrt{\sum_{i=0}^{d-1}[s(k+i) - s(m+i)]^2} \quad (4)$$

For any *v(k)* we can find its nearest neighbor *v(m)* with distance $D(d)_{km}$, increase the dimension to $d+1$ to calculate $D(d)_{km}$, and check expression (5) to determine if the distance meets, which is then a false nearest neighbor.

$$\frac{|D(d)_{km} - D(d+1)_{km}|}{D(d)_{km}} > D_{th} \qquad (5)$$

where D_{th} is the threshold. According to Abarbanel, H.D., Brown, R., Sidorowich, J.J., Tsimring, L.S. the D_{th} lies in the range 10 to 50. In our case we choose D_{th} to be 15. We repeat this procedure from $d = 1$. After each iteration, we increase *d* by 1 until the percent of false nearest neighbors is approaching zero or a small value in case of increasing *d* but the percent of false nearest neighbor decreases slowly or is unchanged.

2.3. Building the model based on Support Vector Regression (SVR)

The goal of a Support Vector Machine (SVM) is to find the optimal hyper plane (Hyper plane may be plane or curve) to classify data into two separate regions so that the distance between the closest point and the hyper plane is at maximum. This is also called the margin. Figure 1 illustrates a hyper plane and a margin.

Assume, the equation of a hyper plane is $\boldsymbol{w}.x + b = 0$. The goal for the SVM algorithm is to find ***w*** and *b* to maximize the margin. The SVM algorithm not only applies to solving classification problems but also to finding solutions to regression subjects. The SVR algorithm is based on a loss function, which is tolerant of error for points distant from the true value with in a small epsilon. This means that this function gives zero error for all the points in training set that lie in the epsilon range. Figures 2 and 3 illustrate linear and nonlinear regression within the epsilon range.

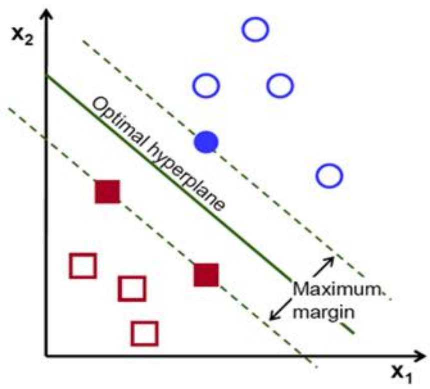


Figure 1. Illustration of the hyper plane and the margin.

Source: Data Mining Map https://www.saedsayad.com/ support_vector_machine_reg.htm

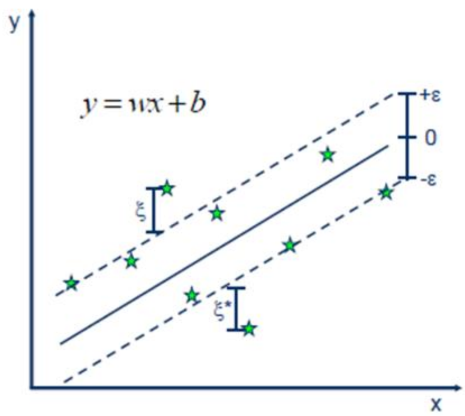


Figure 2. Linear regression with the epsilon range.
Source: Data Mining Map https://www.saedsayad.com/ support_vector_machine_reg.htm

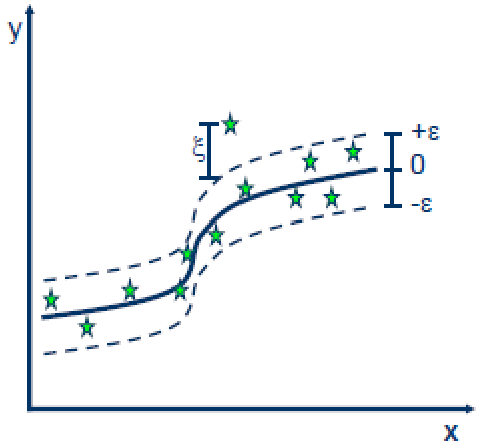


Figure 3. Nonlinear regression with the epsilon range.
Source: Data Mining Map https://www.saedsayad.com/ support_vector_machine_reg.htm

For SVR, the input *x* is mapped into *m* dimension feature space by a nonlinear mapping function first, and then the linear model is built, which is based on this dimension feature space by equation (6):

$$f(x,w) = \sum_{i=1}^{m} w_i \cdot g_i(x) + b \quad (6)$$

where: $g_i(x)$, $i = 1, 2, 3, \ldots, m$ is a set of nonlinear mapping functions.
The accuracy of the estimate is evaluated by loss function $L(y,f(x,w))$. SVR uses a loss function called epsilon – an insensitive loss function which proposed by Vapnik:

$$L = \begin{cases} 0 \ if \ |y - f(x,w)| \le \varepsilon \\ |y - f(x,w)| \ otherwise \end{cases} \quad (7)$$

Thus, SVR is performs linear regression in multi dimension feature space using function *L* and minimizing $\|w\|^2$ for decreasing complexity of the model. This problem can be solved by introducing slug variables ξ_i and ξ_i^* with $i = 1, 2, 3, \ldots n$, to measure the deviation of the training samples which lie outside of the epsilon range. Therefore, SVR is minimized by the function below:

$$min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \quad (8)$$

with constraints:

$$\begin{cases} y_i - f(x_i,w) \le \varepsilon + \xi_i^* \\ f(x_i,w) - y_i \le \varepsilon + \xi_i \\ \xi_i, \xi_i^* > 0 \forall i = 1, \ldots, n \end{cases} \quad (9)$$

Applying the duality theorem for minimizing problems, we finally obtain the function *f(x)*:

$$f(x) = \sum_{i=1}^{n_{sv}}(\alpha_i - \alpha_i^*) \cdot K(x_i,x) + b \qquad 0 \le \alpha_i, \alpha_i^* \le C \quad (10)$$

where: n_{SV} is the number of support vector, and $K(x_i,x)$ is the kernel function which can be defined as

$$K(x_i,x) = \sum_{j=i}^{m} g_j(x_i) \cdot g_j(x) \quad (11)$$

2.4. Building a model bases on fuzzy logic

Assume that we have a data time series that was collected from a system at equal time intervals denoted by *s(1), s(2), s(3), …, s(n)*. The task of the prediction time series to find a mapping from *[s(k-(d-1)T), s(k-(d-2)T, …s(k)) to s(k+T)*, where *d* and *T* are constant positive integer numbers, and *d* is the number of inputs to the predictor, *T* is time lag of time series data. For simple case, we assume $d = 2$ and $T = 1$. Figure 4 below shows the block diagram of the system for prediction. According to the algorithm presented by L. Wang (Li, W., 1994), we first form *n - 2* input – output pairs: *(s(1), s(2) → s(3)), (s(2), s(3) → s(4)), …, (s(n-2), s(n-1) → s(n))*. Next, we find the maximum and minimum of the time series and divide this domain interval into $2*R + 1$ regions (*R* is a positive integer number), denoted by T_1, T_2, …, T_{2*R+1} and then we assign each region with a fuzzy membership function. In our case, we choose the shape of membership function as a triangle wave. Figures 4 and 5 illustrate the fuzzy system and membership function of the input and output with $R = 3$.

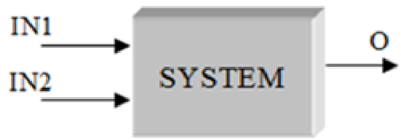


Figure 4. Fuzzy system with 2 input and 1 output.
Source: authors' research

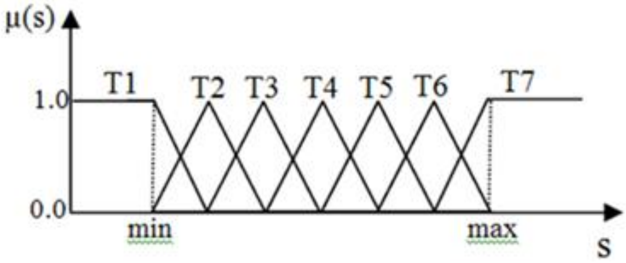


Figure 5. Membership function for input and output.
Source: authors' research

In the next step we calculate the degree of given input-output pairs in different regions, assign it to the region with maximum degree, and then form an *IF – THEN* rule. For example, IN_1 has a max degree 0.85 in region T_1, IN_2 has a max degree 0.7 in region T_3 and O has a max degree 0.8 in region T_7. Hence, we form the rule: **IF** IN_1 is T_1 **AND** IN_2 is T_3 **THEN** O is T_7. Repeating this procedure for each input – output pair gives a set of rules. To avoid conflict rule (two rules have same *IF* part but different *THEN* part), we only accept the rule from the conflict group that has a maximum degree. To do so we use table – lookup to present a fuzzy rule base. The cells of the rule base are filled by the rules. If there is more than one rule on one cell of the fuzzy rule base, then the rule that has highest degree is used. The degree of the rule is calculated by formula (12) below:

$$D(rule) = \mu_A(IN1) \times \mu_B(IN2) \times \mu_C(O) \quad (12)$$

So far, we obtain the fuzzy rule base corresponding to all input-output pairs. The next task is calculating output O when we have a new input sample IN_1 and IN_2. Firstly, calculate the degree of output control of the *k-th* rule combined with the fuzzy rule base corresponding to the new input IN_1 and IN_2 according to formula (13) below:

$$\mu_{O^k}^k = \mu_{I_1^k}(IN1) \times \mu_{I_2^k}(IN2) \qquad (13)$$

where O^k denotes the output region of rule k, and I_j^k denotes the input region of rule k for the *j-th* component.
Finally, we use the centre average defuzzification formula to determine the output.

$$o = \frac{\sum_{k=1}^{N} \mu_{O^k}^k \times o^{-k}}{\sum_{k=1}^{N} \mu_{O^k}^k} \qquad (14)$$

where O^{-k} denotes the center value of region O^k and N is number of rules in the combined fuzzy rule base.

2.5. The Empirical Mode Decomposition method (EMD)

The EMD method is developed from the simple assumption that any non-stationary and non-linear signal consists of number of intrinsic mode of oscillations. In this way, any signal can decomposition into a number of intrinsic mode function (IMF) which has two interesting properties (N. E. Huang, et al., 1998). The first one is in the whole dataset, the number of extrema and the number of zero-crossings must either equal or differ at most by one. The second one is at any point, the mean value of the envelope defined by the local maxima and, the envelope defined by the local minima is zero.

With the above definition for the IMF, the algorithm to decompose any data set x(t) into IFM as flowchart in figure 6.
In figure 6 we should make some clearly describe. The monotonic function is function that has maximum two extrema. The stoppage was used by doctor Huang (Huang et al. 1998). This stoppage criterion is determined by using a Cauchy type of convergence test. Specifically, the test requires the normalized squared difference between two successive sifting operations defined as:

$$SD_k = \frac{\sum_{t=0}^{T} |h_{k-1}(t) - h_k(t)|^2}{\sum_{t=0}^{T} h_{k-1}(t)^2} \qquad (15)$$

If this squared difference SD_k is smaller than a predetermined value, the sifting process will be stopped.

Summing up all IMFs and residue we obtain:

$$x(t) = \sum_{i=1}^{n} IFM_i + r_n \qquad (16)$$

Thus, one achieves a decomposition of the data into n-empirical IMF modes, plus a residue, r_n.

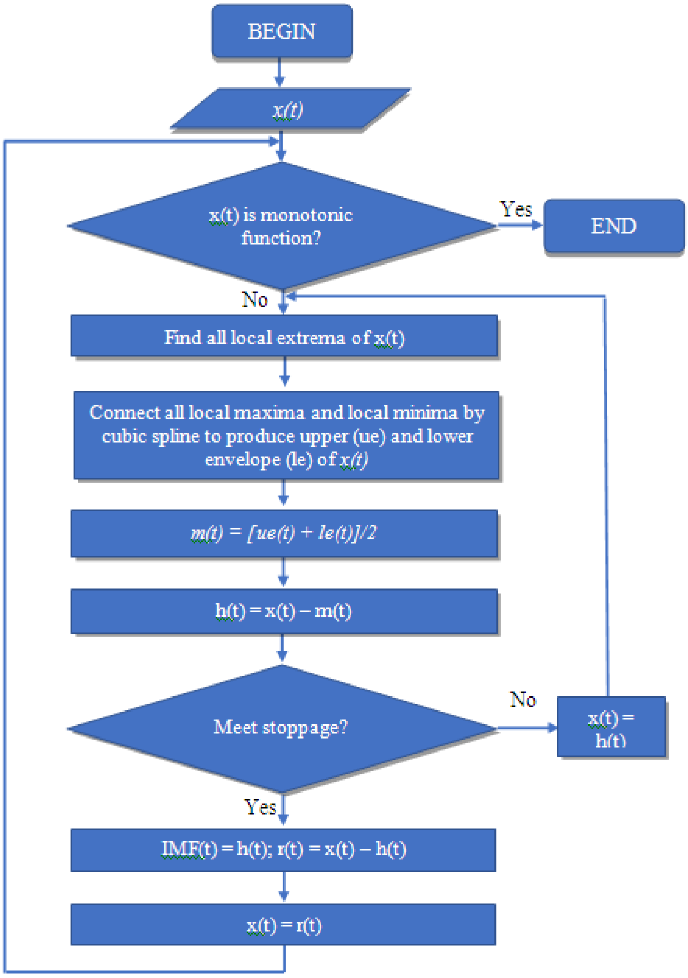
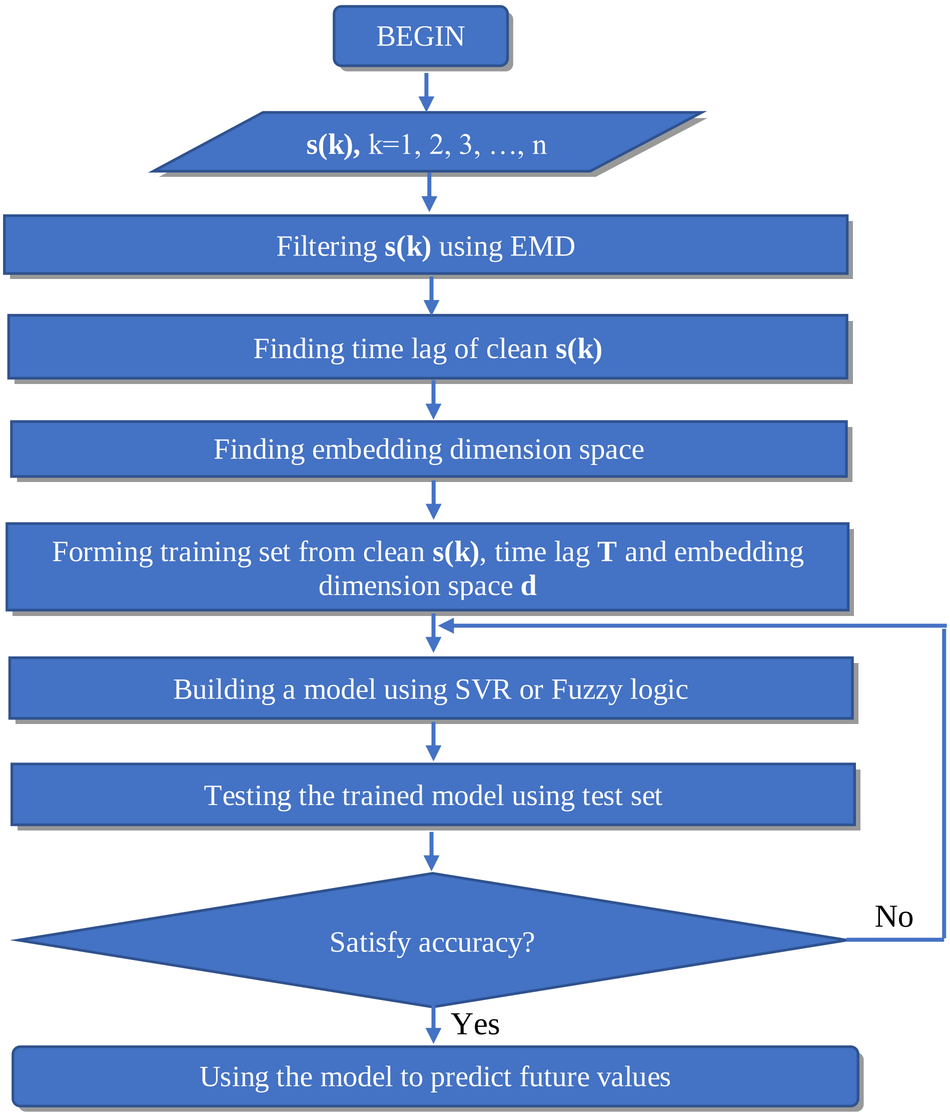


Figure 6. Flowchart of EMD algorithm.
Source: authors' research

III. Principles of our proposal

Principal of our proposed method is illustrated by Figure 7. The first step, we clean the original signal by using EMD. To do so, we decompose original signal into a number of IMFs. After that, we calculate correlation between original signal and each IMF. Finally, we synthesis signal without low correlation and high frequency IMFs. The second step, we find time lag of clean data. The next step, we find embedding dimension space of clean data. After that, forming training set from clean *s(k)*, time lag *T* and embedding dimension space *d*. Then, we train the model is built by fuzzy logic or SVR. Next, we test the trained model using testing data and check the accuracy criterion. If it meets the criterion then, we use the model for predicting future values. Otherwise, we change model's parameters and go back to the training model step.



Figire 7. Principles of our proposed method.
Source: authors' research

IV. Experimental Results

To assess the performance of our proposal we use four time series data consisting of PH, temperature, TSS and COD. These data sets are collected from 0 AM of 25th February 2019 to 2 PM of 26th February 2019 with the sampling period set to five minutes at the waste water processing station of Nittoku paper factory in the Kim Bang district, Ha Nam province, Vietnam. The total data length is 451 points for each parameter. We use the first 401 points as training set and use the remaining 50 points as the test set.

Firstly, each time series data is decomposed into IMFs by the EMD algorithm, and then synthesis signal (clean signal) from IMF and residue without high frequency and low correlation IMF. Figure 8, 9 demonstrate EMD and synthesis for PH time series. After that, time lag and embedding dimension space of clean signal are found. Finally, the models have been built by fuzzy logic or SVR.

With the PH parameter we find that the time lag is one and embedding dimension space is four. This means that we use four points' data from the past to predict one point in the future, or the model has four inputs and one output. Figure 10 shows prediction results by the SVM model vs the fuzzy logic model. The mean square error of SVR is 0.000074. while fuzzy logic is 0.0002.

For the temperature parameter, we found the time lag is two and the embedding dimension space of the time series is four. Figure 11 shows prediction results by the SVM model vs the fuzzy logic model. The mean square error of SVR is 0.008, while fuzzy logic is 0.03.

For the TSS, we found the time lag is 12 and the embedding dimension space of the time series is four. Figure 12 shows prediction results by the SVM model vs the fuzzy logic model. The mean square error of SVR is 0.008, while fuzzy logic is 0.051.

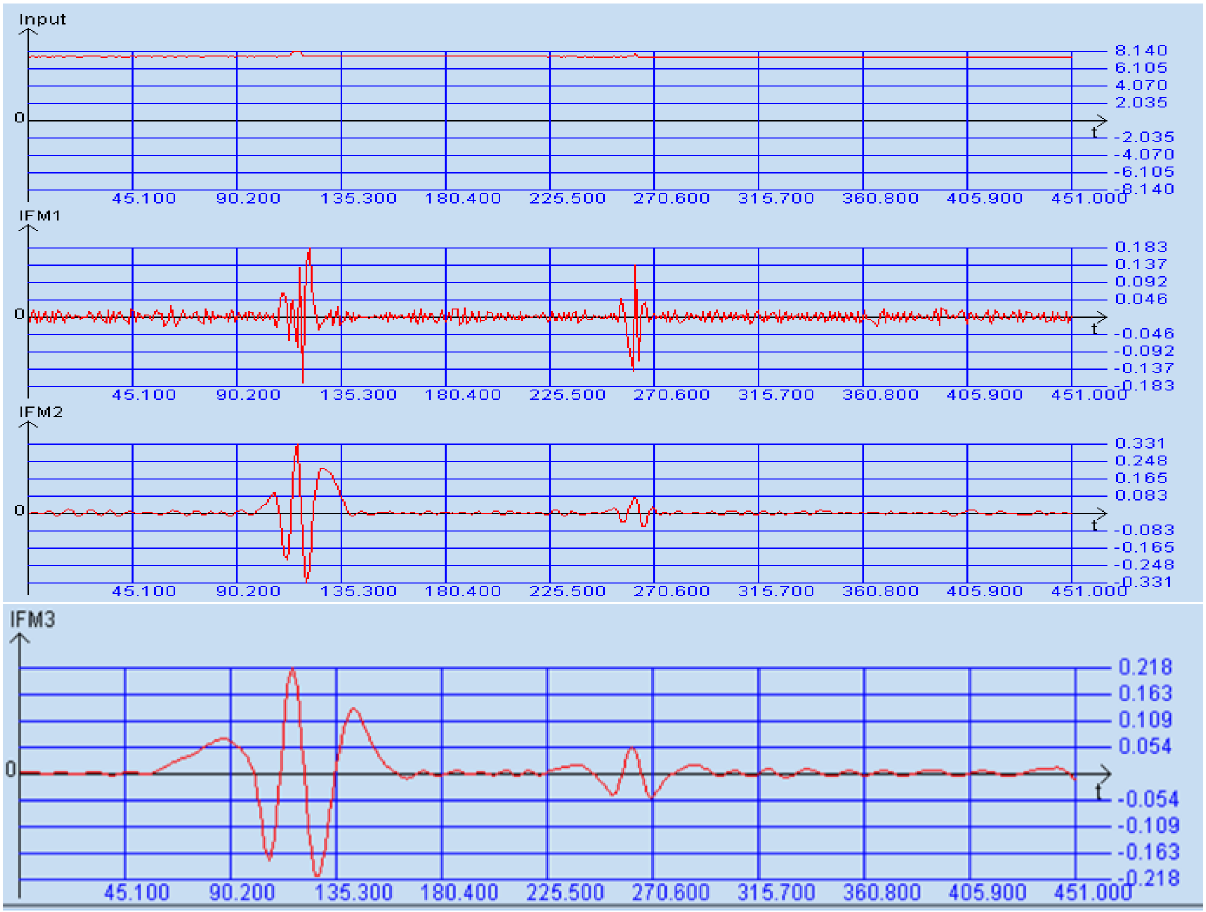
For the COD, we found the time lag is 2 and the embedding dimension space of the time series is five. Figure 13 shows prediction results by the SVM model vs the fuzzy logic model. Mean square error of SVR is 0.018, while fuzzy logic is 0.77.

Table 1 compares the performance between a previous proposed method and the author' novel method

Table 1. Comparison performance between previous and novel method

Method	PH				Temperature				TSS			
	Embed. dim. space	Time lag	MSE		Embed. dim. space	Time lag	MSE		Embed. dim. space	Time lag	MSE	
			Fuzzy	SVR			Fuzzy	SVR			Fuzzy	SVR
Previous	9	1	0.003	0.000077	11	1	0.09	0.04	9	1	0.07	0.06
Novel	4	1	0.002	0.000074	4	2	0.03	0.008	4	12	0.051	0.008

Source: author' research



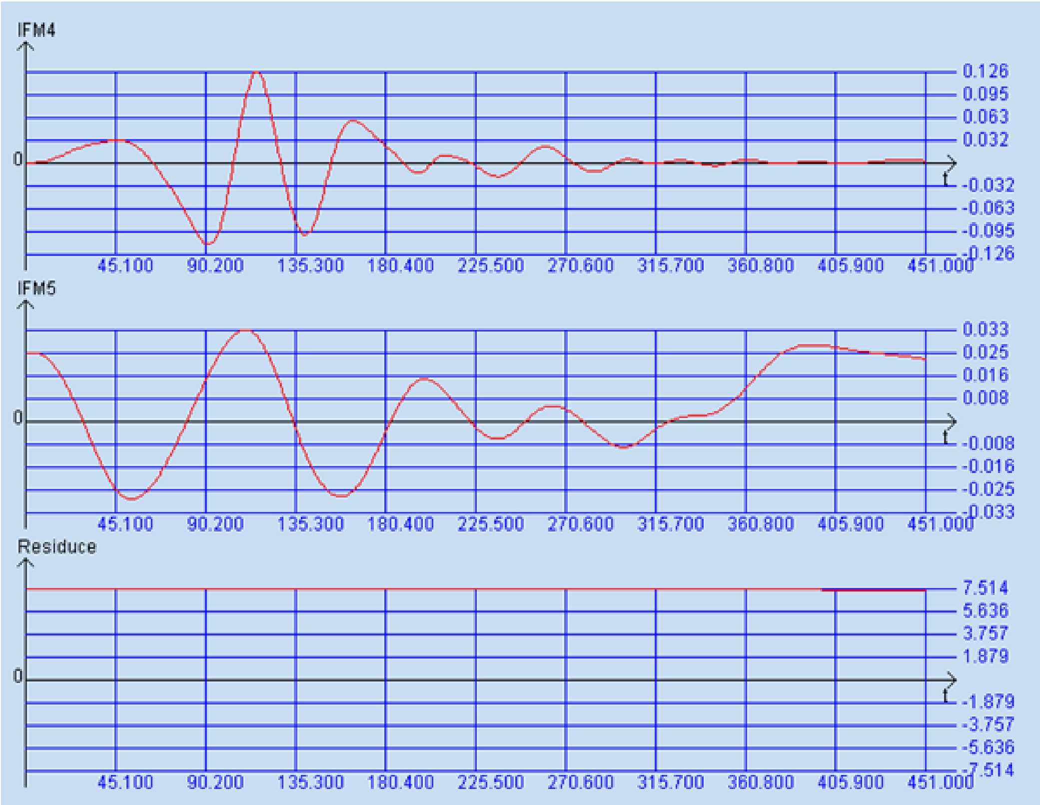


Figure 8. The IMFs of PH signal.
Source: authors' research

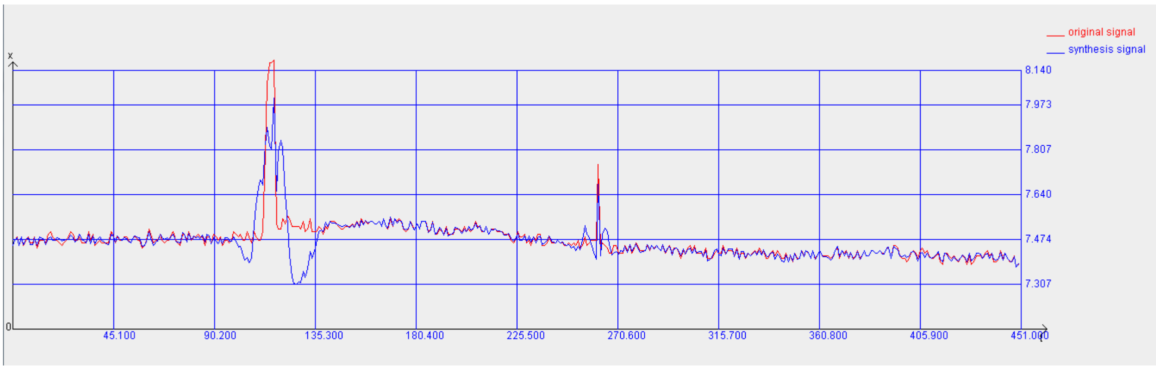


Figure 9. The synthetical PH signal.
Source: authors' research

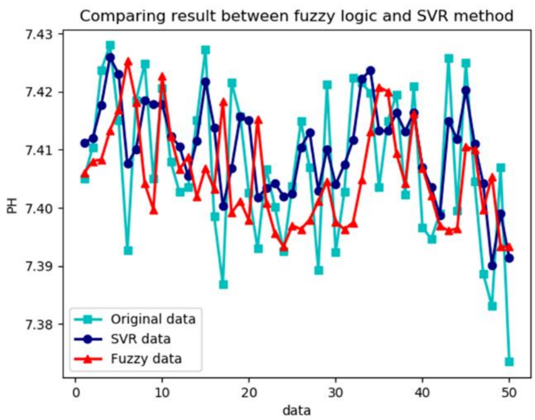


Figure. 10. The prediction result for PH parameter.
Source: authors' research

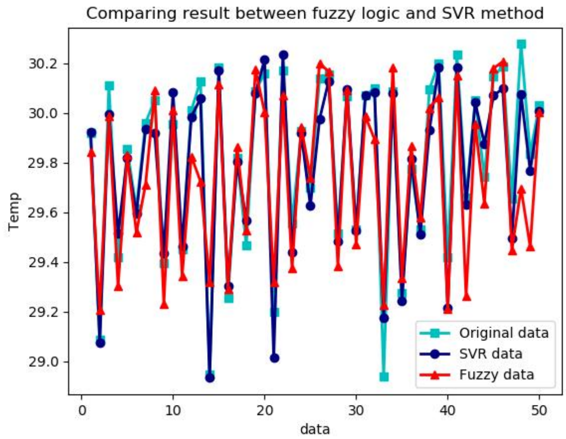


Figure 11. The prediction result for temperature parameter.
Source: authors' research

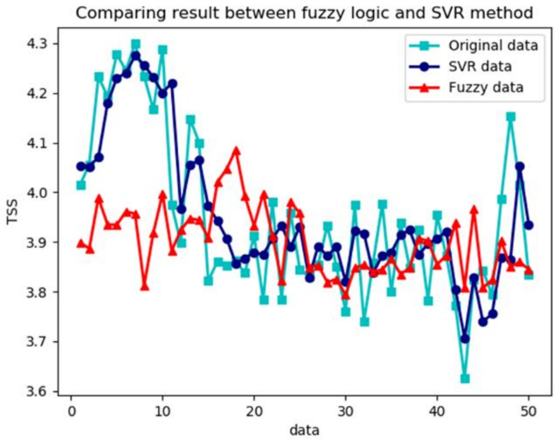


Figure 12. The prediction result for TSS parameter.
Source: authors' research

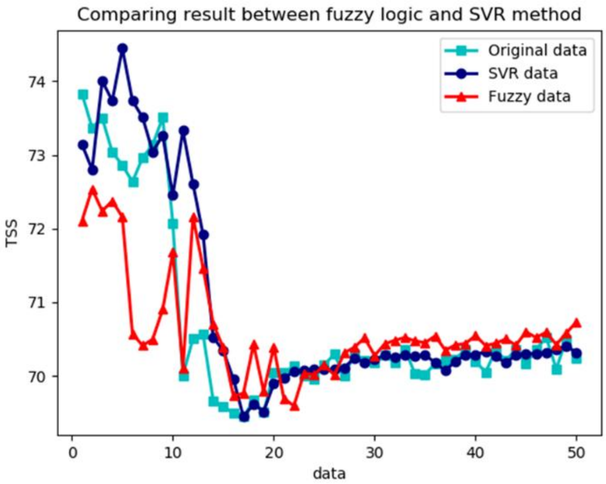


Figure 13. The prediction result for COD parameter.
Source: authors' research

V. Conclusions

In this paper, author present the fusing of empirical mode decomposition, average mutual information, false nearest neighbor algorithm to clean, find time lag and embedding dimension space for time series, fuzzy logic and SVR to build model for predicting quality of waste water in industrial zone. EMD plays adaptive filter role to clean data time series helps decreasing embedding dimension. From the results of experiment show that the proposed method not only decreases complexity of the model but also increases accuracy of prediction. In addition, the experimental results also show that the SVR model obtains the results are more accurate than fuzzy model for all case study. However, training fuzzy logic model spend smaller time than SVR model because training fuzzy logic model we only need to travel training data set one time while training SVR model we need to travel training data set many times. Therefore, we can apply fuzzy logic model for real time and least accurate parameter. Otherwise, we apply SVR model.

References

[1]. According to Circular 24 of the Vietnamese Ministry of Natural Resources and Environment: 24/2017/TT-BTNMT, 1st September 2017, Ha Noi, Vietnam, 2017

[2]. Abarbanel, H.D., Brown, R., Sidorowich, J.J., Tsimring, L.S., (1993). The analysis of observed chaotic data in physical systems. *Reviews of modern physics*. 1993 Oct 1;65(4):1331.

[3]. Božić, J., Vukotić, S., Babić, D., (2011). Prediction of the RSD exchange rate by using wavelets and neural networks. In 2011 19th *Telecommunications Forum (TELFOR) Proceedings of Papers* 2011 Nov 22 (pp. 703-706). IEEE.

[4]. Castillo, O., Melin, P. (2002). Hybrid intelligent system for time series prediction using neural networks, fuzzy logic, and fractal theory. *IEEE transactions on neural network* Vol.13, No.6, November 2002, https://doi.org/10.1109/TNN.2002.804316.

[5]. Frank, R.J., Davey, N., Hunt, S.P., (2001). Time series prediction and neural networks. *Journal of intelligent and robotic systems*. 2001 May 1;31(1-3):91-103.

[6]. Hanias, M.P., Curtis, P.G. (2008). Time series prediction of dollar\euro exchange rate index. *International Research Journal of finance and economics*. 2008;15:232-9.

[7]. Hua, X., Zhang, D., Leung, S.C. (2010). Exchange rate prediction through ANN Based on Kernel Regression. In 2010 Third International Conference on *Business Intelligence and Financial Engineering* 2010 Aug 13 (pp. 39-43). IEEE, https://doi.org/10.1109/BIFE.2010.20.

[8]. Iokibe, T., Murata, S., Koyama, M., (1995). Prediction of foreign exchange rate by local fuzzy reconstruction method. *In 1995 IEEE International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century* 1995 Oct 22 (Vol. 5, pp. 4051-4054). IEEE

[9]. Wang, Y.-F. (2002). Predicting stock price using fuzzy grey prediction system. *Expert system with application* 22 (2002) 33-39, https://doi.org/10.1016/S0957-4174(01)00047-1

[10]. Lanzhen, L., Tan, S. (2010). The study on short-time wind speed prediction base on time-series neural network algorithm. *Power and energy engineering conference* (APPEEC), 2010 Asia-Pacific

[11]. Liu, F-Y., (2010). Exchange Rate Based on Wavelet and Support Vector Regression. *Advanced Computer Control* (ICACC), 2010 2nd International Conference.

[12]. Liu, W., (2008). Forecasting Exchange Rate Change between USD and JPY Using Dynamic Adaptive Neuron-Fuzzy Logic System. *Asia Pacific Journal of Finance and Banking Research*. 2008 Jul 4;2(2)

[13]. Li, W., (1994). Adaptive Fuzzy Systems and Control Design and Stability Analysis. *Prentice Hall International*, 1994, pp. 65-69.

[14]. N. E. Huang, et al., (1998). The empirical mode decomposition and the Hilbert spectrum for non-linear and non-stationary time series analysis, *Proceedings of the Royal Society of London Series A—Mathematical Physical and Engineering Sciences*, 1998, 454: 903-995.

[15]. Nghien Nguyen Ba, Ricardo Rodriguez Jorge, (2019). Building an Early Warning Model for Detecting Environmental Pollution of Wastewater in Industrial Zones. Advances on P2P, Parallel, Grid, Cloud and Internet Computing, *Proceedings of the 14th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing* (3PGCIC-2019 pp. 828-837)

[16]. Support Vector Regression – Data Mining Map https://www.saedsayad.com/ support_vector_machine_reg.htm

[17]. Thiang, Y.K. (2010) Electrical load time series data forecasting using interval type 2 Fuzzy logic system. In 2010 3rd International Conference on *Computer Science and Information Technology*, vol. 5, pp. 527-531. IEEE, 2010, https://doi.org/10.1109/ICCSIT.2010.5563903.

[18]. T. Do Duc, N. Nguyen Ba, C. Nguyen Van, T. Nguyen Nhu, D. Hoang Tien (2020) Surface Roughness Prediction in CNC Hole Turning of 3X13 Steel using Support Vector Machine Algorithm, Tribology in Industry, http://www.tribology.rs/journals/aips/940.pdf, DOI:10.24874/ti.940.08.20.11.

[19]. NusratParveen, SadafZaidi, MohammadDanish (2016), Support vector regression model for predicting the sorption capacity of lead, Perspectives in Science: Volume 8, September 2016, Pages 629-631, https://doi.org/10.1016/j.pisc.2016.06.040.

[20]. Muthukrishnan. R, Maryam Jamila. S (2020), Predictive Modeling Using Support Vector Regression, International Journal of Scientific & Technology Research volume 9, issue 02, february 2020, http://www.ijstr.org/final-print/feb2020/Predictive-Modeling-Using-Support-Vector-Regression.pdf.

Dung Vuong Quoc, et. al. "Fusion of Soft Computing and Chaos Theory to Build A Model For Prediction Quality Of Wastewater In Industrial Zones." *IOSR Journal of Computer Engineering (IOSR-JCE)*, 24(1), 2022, pp. 31-41.