A Fusion Method to Build A Model for Prediction Exchange Rate from USD to VND

Hai Nguyen Thanh*
Information Technology faculty
Hanoi University of Industry
Hanoi, Vietnam

Anh Pham The
Information Technology faculty
Hanoi University of Industry
Hanoi, Vietnam

Nhung Tran Phuong
Information Technology faculty
Hanoi university of Industry
Hanoi, Vietnam

Thang Le Anh
Information Technology faculty
Hanoi university of Industry
Hanoi, Vietnam

Truong Do Sinh
Information Technology faculty
Hanoi university of Industry
Hanoi, Vietnam

Phu Nguyen Trung
Information Technology faculty
Hanoi university of Industry
Hanoi, Vietnam

***Abstract*—In this paper, the combination of the Hilbert-Huang Transform, Fuzzy logic system and an embedding theorem is described to predict the short-term exchange rate from United States dollar to Vietnamese Dong. By using the Hilbert-Huang Transform as an adaptive filter, the proposed method decreases the embedding dimension space from twelve (original samples) to four (de-noising samples). This dimension space provides the number of inputs to the fuzzy rule base system, which causes the number of rules, the time for training and the inference process to reduce. Experimental results indicated that this method not only reduces complication of the model but also achieves higher accuracy prediction than the direct use of original data.**

Keywords*—Embedding dimension space; average mutual information; Fuzzy rule base system; prediction; HHT

I. INTRODUCTION

Forecasting financial time series serves an important role in daily life, especially the exchange rate forecast that helps an importer and exporter to choose the best time to import or export products to obtain the highest profit. Researchers have successfully employed approaches for predicting exchange rates. In [1], authors combined kernel regression (KR) and the Function Link Artificial Neural Network (FLANN) to predict the exchange rate from United States dollar (USD) to British Pound (GBP), Indian Rupee (INR) and Japanese Yen (JPY). KR served a role in filtering, and FLANN was a model for prediction. The authors in [2] use chaos theory and reconstructed state space for predicting the exchange rate between USD and EURO (EUR). Fan-Yong Liu [3] uses a hybrid discrete wavelet transform (DWT) and support vector regression (SVR) to predict the exchange rate between Chinese Yuan (CNY) and USD. First, this researcher uses DWT to decompose time series data to different time scales and later appropriately chooses a kernel function for SVR and a prediction that corresponds with each time scale. Her synthesis prediction is obtained from different predicted time scale results. The authors in [4] use a local fuzzy reconstruction method to predict the exchange rate between JPY, USD and Canadian dollar (CAD). The authors in [5] use a successful wavelet transform to filter noise in the exchange rate time series before using it to train and predict the base on the Multi-Layer Feed Forward Neural Network. Weiping Liu [6] uses a hybrid neuron and fuzzy logic to predict the exchange rate between JPY and USA. In 1998, Dr. Huang proposed a novel method for decomposition nonlinear and nonstationary time series into a set of multi-time scale signals, which are referred to as intrinsic mode function (IMF). This method is successfully applied in many fields, such as an adaptive filter for nonstationary and nonlinear data [7], [8]. In this paper, the combination of HHT, Fuzzy logic system, average mutual information and an embedding theorem to predict the exchange rate from USD to VND is presented. This method is simple, adaptive and high-accuracy prediction.

This paper is organized as follows: In the next section, related studies are presented. Section 3 discusses the principle of the proposed method. Section 4 demonstrates the accuracy prediction of the method by experiment. Section 5 presents the conclusion and future studies.

II. RELATED WORK

A. Finding the time delay

According to the literature review [11], If we select too small time delay T, then two data points $s(n+jT)$ and $s(n+(j+1)T)$ will be so close to each other that we cannot distinguish them from each other. Similarly, if we choose so large T, then $s(n+jT)$ and $s(n+(j+1)T)$ are completely independent of each other in a statistical sense. To determine the 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 systems denoted by a_k, b_k, the mutual information between a_k and b_k is specified as equation (1) below.

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

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 $P_{AB}(a, b)$.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 09, September-2020

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:

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

To apply this definition into time series data *s(n)* which is measured from a physical system. We consider the 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).

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

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.

B. Finding the time delay

The method of false nearest neighbors (FNNs) has been proposed in [11] to obtain the minimum embedding dimension. The principle of the method is based on the idea that points that are close to each other may not be neighbors even if the embedding dimension is increased. The FNNs method is used to calculate the adequate number of dimensions for embedding a time series. For a given time series, the data comprises *y(k)*, where *k = 1, 2, …, n*. The idea of the method is to combine sequence values into vectors and construct *d*-dimensional vectors from the observed data using a delay embedding as shown in Eq. (4) [11][12].

$$\mathbf{Y}(k) = [\mathbf{y}(k), \mathbf{y}(k+\tau), \cdots, \mathbf{y}(k+(d-1)\tau)] \quad (4)$$

$k = 1, 2, \ldots N-(d-1)$

Each vector *y(k)* has a nearest neighbor $y^{NN}(k)$ with nearness in the sense of some distance function, in dimension *d*. For each vector, its nearest neighbor is obtained in *d*-dimensional space using the Euclidean distance in Eq. (5).

$$D_d(k) = \sqrt{\sum_{i=0}^{d-1}[\mathbf{y}(k+i\cdot\tau) - \mathbf{y}^{NN}(k+i\cdot\tau)]^2} \quad (5)$$

Next, the distance between the vectors in *d*-dimensional space is compared with the distance between the vectors when embedded in dimension *d + 1*, as shown in Eq. (6).

$$\frac{|\mathbf{y}(k+i\cdot\tau) - \mathbf{y}^{NN}(k+i\cdot\tau)|}{D_d(k)} > R_t \quad (6)$$

where R_t represents the threshold. In [11], the authors recommend the range $10 \le R_t \le 50$. In our case, $R_t = 10$ and a second criterion of falseness of nearest neighbors has been considered as suggested in [11] (refer to Eq. (7)).

$$\frac{D_{d+1}(k)}{R_a} \ge A_t \quad (7)$$

where R_a is the standard deviation of the given time series data, and $A_t = 2$. For instance, $\boldsymbol{y}(k)$ and its nearest neighbors are false nearest neighbors if either Eq. (7) or Eq. (8) fails.

C. Hilbert-Huang Transform

The Hilbert-Huang Transform is proposed by Dr. Huang in 1998 [9] and consists of two parts. The key part is empirical mode decomposition (EMD). In this part, each signal is decomposed into a finite set number of Intrinsic Mode Function (IMFs), which satisfies two criteria [9]:

The first one is that the number of extremes and zero crossings must be equal or differ at most by one in the whole data set.

The second one is that this number is symmetrical, which indicates that the mean of the upper envelope at any point connects all local maxima and the lower envelope that connects all local minima is zero.

The flowchart of the EMD algorithm to decompose any signal $\boldsymbol{x}_1(k)$ to IMFs is illustrated in Figure 1. Three stopping criteria exist. The first criterion was employed by Dr. Huang in 1998 [9]. This stopping criterion is determined using a Cauchy type of convergence test. The test requires the normalized squared difference between two successive sifting operations, which are defined as follows:

$$SD_i = \frac{\sum_{k=0}^{N}\left|\mathbf{x}_{1_{i-1}}(k) - \mathbf{x}_{1_i}(k)\right|^2}{\sum_{k=0}^{N}\mathbf{x}_{1_{i-1}}^2(k)} \quad (8)$$

For the given small threshold (th) value, the sifting process will stop when SD_i is less than a small chosen threshold th. For the second criterion, the sifting process will only stop after S consecutive times when the numbers of zero-crossings and extremes remain the same and are equal or differ at most by one. S is the predefining value; its optimal value ranges from four to eight as suggested by Dr. Huang [8][9]. The criterion has also been suggested by Dr. Huang: the number of shifts should be fixed at ten. In our case, the first criterion is applied.

After the EMD process, a set of IMFs is obtained from high frequency to low frequency oscillation and residue (trend). Summing all IFMs and residue, the original signal is obtained.

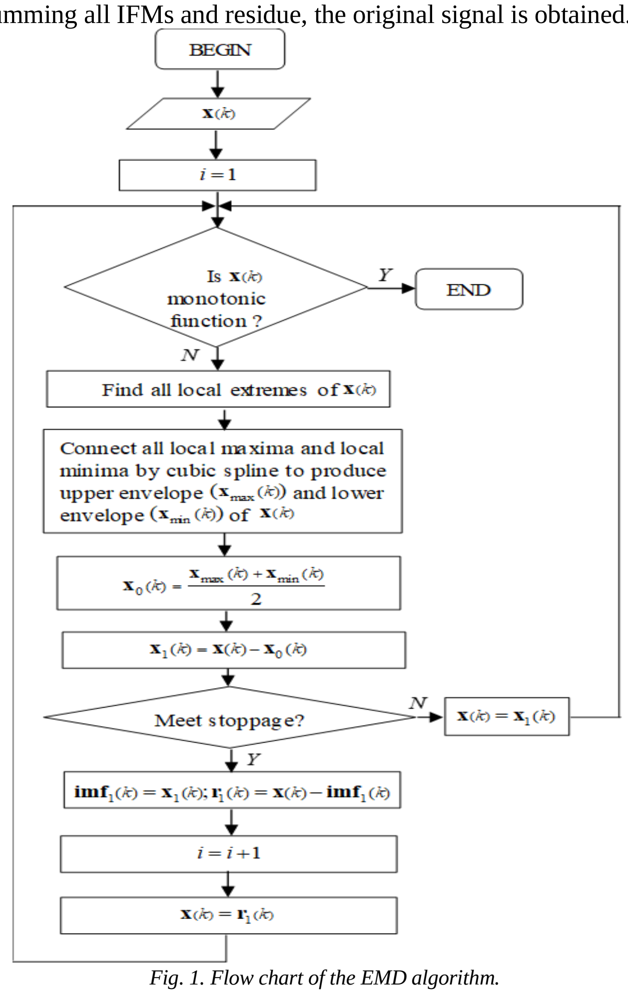

Fig. 1. Flow chart of the EMD algorithm.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 09, September-2020

D. Building the model based on Fuzzy rules base system

Assume a data time series $\boldsymbol{y}(k)$ that has been collected from a system at equal time intervals denoted by $y(1), y(2), y(3), \ldots, y(N)$. The task of the prediction time series is to obtain finding a mapping from $[y(k-n+1), y(k-n+2), \ldots, y(k)]$ to $y(k+1)$, where n and l are constant positive integer numbers, and n is the number of inputs to the predictor. For a simple case, we assume $n = 2$ and $l = 1$. Figure 2 shows the block diagram of the system for prediction. According to the algorithm presented by L. Wang[10], we form $n-2$ input–output pairs $(y(1), y(2) \rightarrow y(3))$, $(y(2), y(3) \rightarrow y(4))$, $\cdots, (y(N-2), y(N-1) \rightarrow y(N))$.

***Fig. 2**. Block diagram of the system.*

Next, we obtain the maximum and minimum of the time series and divide this domain interval into $2 \times R + 1$ regions (R is a positive integer number) denoted by $T1, T2, \ldots, T2R, T2R+1$ and assign each region with a fuzzy membership function. In our case, we choose the shape of the membership function a triangle. Figure 3 illustrates the membership function of input and output with $R = 3$.

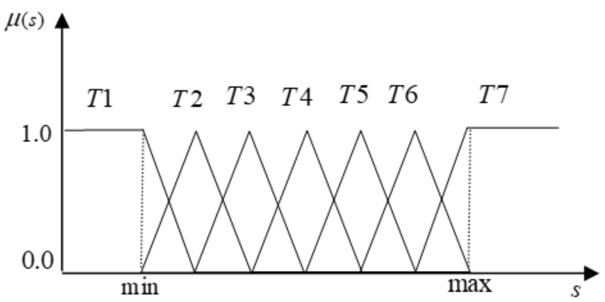

***Fig. 3**. Membership function of the input and output.*

In the next step, we calculate the degree of a given input-output pair in different regions and assign it to a region with a maximum degree and form **IF – THEN** rules. For example, $IN1$ has a maximum degree of 0.8 in region $T1$, $IN2$ has a maximum degree of 0.5 in region $T4$ and O has a maximum degree of 0.9 in region $T7$. Therefore, we form the following rule: **IF** $IN1$ is $T1$ **AND** $IN2$ is $T4$ **THEN** O is $T7$. To repeat this procedure for each input – output pair, we obtain a set of rules. To avoid the conflict rule (two rules have the same IF part but different THEN parts), we only accept the rule from the conflict group that has the maximum degree. We use table – lookup to present a fuzzy rule base. The cells of the rule base are filled by the rules; if more than one rule exists on one cell of the fuzzy rule base, the rule with the maximum degree is applied. The degree of the rule is calculated by Eq. (9).

$$D(rule) = \mu_A(IN1) \times \mu_B(IN2) \times \mu_C(O) \quad (9)$$

We obtain the fuzzy rule base that corresponds to all input-output pairs. The next task is to calculate the output O when we have new input samples $IN1$ and $IN2$. We first calculate the degree of output control of rule $k-th$ in the combined fuzzy rule base that corresponds to the new inputs $IN1$ and $IN2$ according to Eq. (10).

$$\mu_{O^k}^k = \mu_{I_1^k}(IN1) \times \mu_{I_2^k}(IN2) \quad (10)$$

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. We use the centre average defuzzification equation to determine the output, given by Eq. (11).

$$o = \frac{\sum_{k=1}^{N} \mu_{O^k}^k \times o^{-k}}{\sum_{k=1}^{N} \mu_{O^k}^k} \quad (11)$$

where O^{-k} denotes the center value of region O^k, and N is the number of the rule.

III. PRINCIPLE OF THE PROPOSED METHOD

The principal of our proposal method is illustrated by Figure 4. Firstly, we decompose a signal into IMFs. Secondly, we reconstruct the signal without IMF1 (highest frequency oscillation) to reduce noise. After that, we find the time lag and embedding dimension of the de-noised signal. Next, we use the de-noise time series and obtained the time lag and embedding dimension to create a model using SVR. After that we test the accuracy of the model by using test set. We use MSE (mean square error) as performance index. If this index less than or equals small predefined value then we move to the final step. Otherwise, we change parameters (C, ε) of the model and train the model again.

Finally, we use the tested model to predict the future value of the exchange rate between USD and VND.

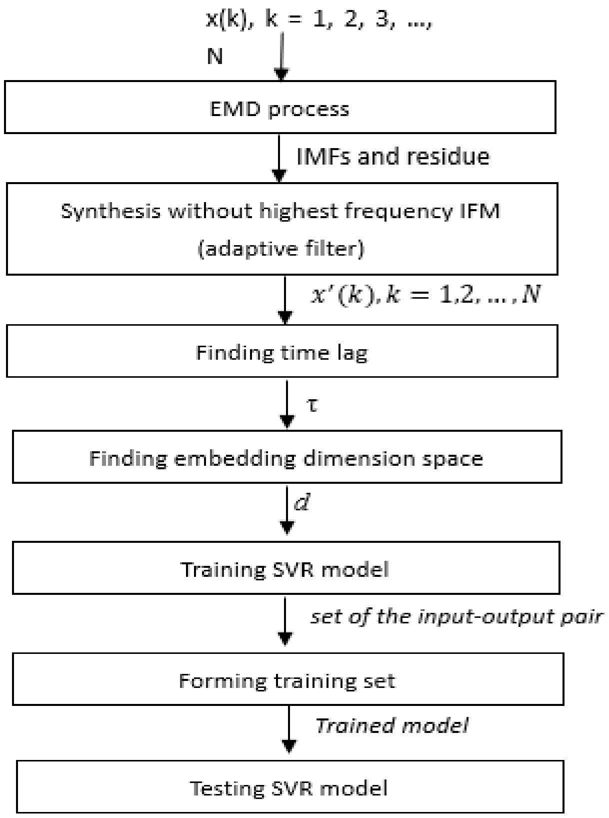

Fig. 4. Principle of our proposed method.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 09, September-2020

IV. EXPERIMENTAL RESULTS

To assess the performance of our proposal, we use the daily exchange rate between USD and VND from January 1, 2019 to December 31, 2019 (https://vn.investing.com/currencies/usd-vnd-historical-data). The total data length of the data set is 261 samples. We divide data set into two sets. The training set consists of the first 221 data points and the test set has the last 40 data points. Figure 5 shows the original data vs. the de-noises data using HHT.

Figures 6 and 7 illustrate the number of false nearest neighbor of the original time series and the de-noise time series.

Figure 8 shows the testing results produced from the model that be trained by the original data and the filtered data.

Table 1 compares the performance between the model using the original time series data and the model using the de-noise time series data.

The embedding dimension of the original time series is twelve, whereas the embedding dimension of the de-noised time series is four. Decreasing the embedding dimension causes a decrease in the complexity of the dynamic system.

In our case, we decrease the number of inputs to the Fuzzy logic system. This phenomenon not only reduces the training time but also decreases the prediction time and increase the accuracy of prediction.

Fig. 5. The de-noise time series vs the original time series.

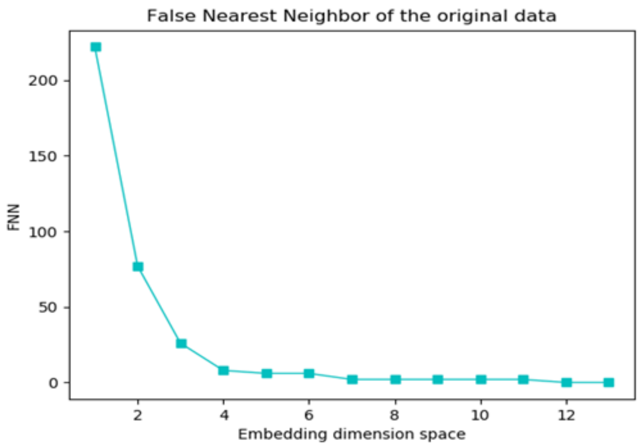

Fig. 6. The embedding dimension space of the original time series.

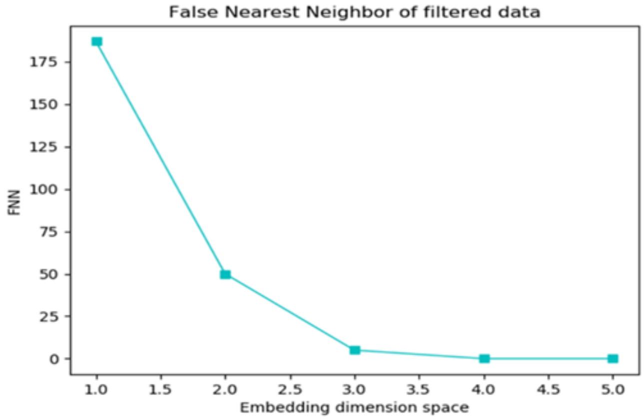

Fig. 7. The embedding dimension space of the de-noise time series.

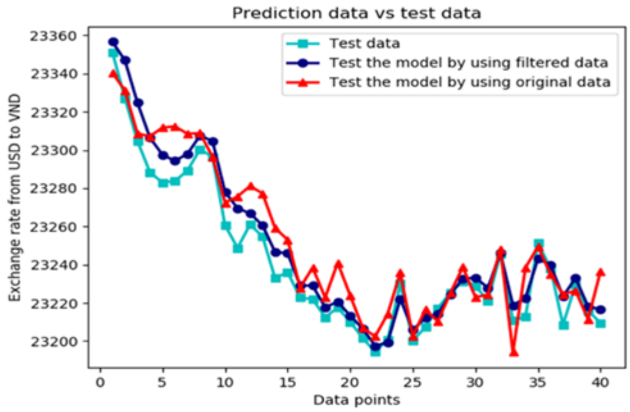

Fig. 8. Comparing the performance of the model using the direct time series and the model using the de-noise time series.

V. CONCLUSION

We presented the combination of HHT and the algorithm of false nearest neighbor to obtain a minimum embedding dimension space, average mutual information to find the time lag of time series and fuzzy rules base system to predict the exchange rate between USD and VND. The HHT serves a role in adaptive filters and reconstructed signals without high-frequency oscillation (IMF1) can reduce noise. We use a de-noised signal to obtain the time lag, and the embedding dimension space and the training fuzzy logic model. The experiment revealed that the de-noised signal can decrease the embedding dimension space, decrease the complexity of the system and achieve higher accuracy prediction than the direct use of the original signal.

In the future, we are going to apply more soft computing techniques such as Feed Forward Neuron Networks (FFNN), Support vector regression to build the model for prediction exchange rate, compare the accuracy prediction among models to choose the best model for prediction exchange rate between USD and VND.

TABLE I. RESULTS OF PREDICTION USING THE ORIGINAL AND THE DE-NOISE TIME SERIES

Data point	***Original data***	***Pre. result using the original data***	***Pre. result using the de-noise time series***	***Square error of using the original data***	***Square error of using the de-noise data***
x1	23351.25	23340.45	23356.63	116.64	28.94
x2	23327.06	23331.19	23346.96	17.06	396.01
x3	23304.76	23308.66	23324.83	15.21	402.80
x4	23288.36	23307.16	23306.46	353.44	327.61
x5	23282.97	23311.71	23297.51	825.99	211.41
x6	23283.80	23312.32	23294.28	813.39	109.83
x7	23289.17	23308.50	23297.97	373.65	77.44
x8	23300.41	23308.81	23307.53	70.56	50.69
x9	23296.39	23296.47	23304.66	0.01	68.39
x10	23260.61	23272.43	23278.11	139.71	306.25
x11	23248.78	23275.66	23269.49	722.53	428.90
x12	23261.03	23281.18	23266.86	406.02	33.99
x13	23254.68	23277.23	23260.53	508.50	34.22
x14	23232.90	23259.10	23246.55	686.44	186.32
x15	23236.11	23253.09	23246.01	288.32	98.01
x16	23222.72	23227.96	23229.18	27.46	41.73
x17	23221.94	23238.44	23228.98	272.25	49.56
x18	23212.49	23222.75	23217.71	105.27	27.25
x19	23217.54	23240.95	23220.58	548.03	9.24
x20	23210.06	23223.90	23213.29	191.55	10.43
x21	23201.58	23206.89	23206.56	28.20	24.80
x22	23194.25	23202.44	23197.41	67.08	9.99
x23	23200.90	23214.33	23199.44	180.36	2.13
x24	23230.12	23235.74	23222.14	31.58	63.68
x25	23200.25	23202.52	23205.80	5.15	30.80
x26	23207.25	23216.41	23212.18	83.91	24.30
x27	23217.21	23210.30	23214.02	47.75	10.18
x28	23225.34	23225.16	23224.60	0.03	0.55
x29	23231.21	23238.74	23232.60	56.70	1.93
x30	23228.92	23222.98	23233.27	35.28	18.92
x31	23221.02	23224.27	23227.88	10.56	47.06
x32	23245.78	23248.19	23245.99	5.81	0.04
x33	23210.95	23194.21	23218.77	280.23	61.15
x34	23212.69	23238.43	23222.60	662.55	98.21
x35	23251.27	23249.43	23243.15	3.39	65.93
x36	23238.08	23235.15	23239.81	8.58	2.99
x37	23208.57	23224.45	23223.33	252.17	217.86
x38	23229.50	23226.45	23233.03	9.30	12.46
x39	23215.91	23211.40	23218.09	20.34	4.75
x40	23209.51	23236.58	23216.61	732.78	50.41
			Mean square error	**225.08**	**91.18**

ACKNOWLEDGMENT

This work is supported by the Hanoi University of Industry.

REFERENCES

[1] Hua, X., Zhang, D., Leung, S.C.H.: Exchange rate prediction through ANN based on Kernel Regression. 2010 Third International Conference on Business Intelligence and Financial Engineering. August 13-15, 2010, 39-43

[2] Hanias, M. P., Curtis, P. G.: Time Series Prediction of Dollar\Euro Exchange Rate Index. International Research Journal of Finance and Economics. May 2008, Issue 15, 224-231

[3] Liu, F.-Y.: The Hybrid Prediction Model of CNY/USD Exchange Rate Based on Wavelet and Support Vector Regression. 2010 2nd International Conference on Advanced Computer Control (ICACC). March 27-29, 2010, 561 - 565

[4] Iokibe, T., Murata, S., Koyama, M.: Prediction of Foreign Exchange Rate by Local Fuzzy Reconstruction Method. Systems, Man and Cybernetics, 1995. IEEE International Conference on Intelligent Systems for the 21st Century, vol.5, Oct. 22-25, 1995, 4051 – 4054

[5] Božic, J., Vukotic, S., Babic, Đ.: Prediction of the RSD exchange rate by using wavelets and neural networks. 2011 19th Telecommunications forum (TELFOR), Nov. 22-24, 2011, 703 – 706

[6] Liu, W.: Forecasting exchange rate change between usd and jpy by using dynamic adaptive neuron-fuzzy logic system. Asia Pacific Journal of Finance and Banking Research. 2 (2) (2008) 1-12

[7] Zhang, L., Wu, D., Zhi, L.: Method of removing noise from EEG signals based on HHT method. The 1st International Conference on Information Science and Engineering (ICISE 2009). Dec. 26-28, 2009, 596 – 599

[8] Huang, N. E., Wu, M.-L., Qu, W., Long, S. R., Shen, S. S. P.: Applications of Hilbert–Huang transform to non-stationary financial time series analysis. Applied stochastic models in business and industry. 19 (3) 245–268

[9] Huang, N. E., Shen, S. S. P.: Hilbert-Huang Transform and Its Application. World Scientific Pub Co Inc, 2005

[10] Wang, L.-X.: Adaptive fuzzy systems and control: Design and stability analysis. Prentice Hall International. Cliffs, NJ. (1994) pp. 65-69

[11] Ababarnel, H.D.I., Brown, R., Sidorowich, J.J., Tsimring, L.S.: The analysis of observed chaotic data in physical systems. Reviews of Modern Physics. 65 (4) (1993) 1331-1392

[12] Rodríguez, R., Bila, J., Mexicano, A., Cervantes, S., Ponce, R., Nghien, N. B.: Hilbert-Huang transform and neural networks for electrocardiogram modeling and prediction. In 2014 10th International Conference on Natural Computation (ICNC). IEEE. (2014) 561-567