

## EUR/RSD Exchange Rate Forecasting Using Hybrid Wavelet-Neural Model: A CASE STUDY

Jovana Božić<sup>1</sup> and Đorđe Babić<sup>1</sup>

<sup>1</sup> School of Computing, Knez Mihailova 6/6,  
11000 Belgrade, Serbia  
jovanabozic@gmail.com, djbabac@raf.edu.rs

**Abstract.** In this paper, we examine and discuss modeling and prediction results of several exchange rates, with main focus on EUR/RSD, using a combination of wavelet transforms, neural networks and statistical time series analytical techniques. We have also designed a user friendly software prediction tool in MATLAB which implements the proposed model. The analyzed hybrid model combines the capabilities of two different wavelet transforms and neural networks that can capture hidden but crucial structure attributes embedded in the exchange rate. The financial time series is decomposed into a wavelet representation using two different resolution levels. For each of the new time series, a neural network is created, trained and used for prediction. In order to create an aggregate forecast, the individual predictions are combined with statistical features extracted from the original input. Additional to the conclusion that the increase in resolution level does not improve the prediction accuracy, the analysis of obtained results indicates that the suggested model sufficiently satisfies characteristics of a financial predictor.

**Keywords:** Time-series forecasting, wavelet packet transform, stationary wavelet transform, neural networks.

### 1. Introduction

Due to the rapid expansion of global trading markets over the last few decades, the currency exchange market has experienced remarkable growth [1]. As a consequence, in today's interlinked global economy, exchange rates can have a positive and/or negative effect on the rising level of imports and exports. Therefore, the prediction of exchange rate is a crucial factor for the controlling of the export-import markets and for the success of a country's financial institutions and businesses operating in it.

In this paper, we use a time series prediction using hybrid wavelet-neural model. In order to expose the complex underlying structures for deeper evaluation, the time-series is first subjected to a wavelet-based decomposition process using decomposition levels of two and three. The decomposed signal components are then used as input elements to a new group of neural networks where valuable information is captured during the process of the training phase. In the third stage of the model, the newly obtained (simulated) time series along with the statistical features extracted from the original input are directed into the final neural network where the prediction is made (third

stage). Based on the proposed hybrid model, we have designed a user friendly MATLAB program. This program is a very convenient tool for prediction of time-series of this type. Furthermore, it allows user to change easily several system parameters and to perform statistical analysis of obtained results. Comparison of the prediction results of the two models are based on four evaluative parameters: MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error), MSE (Mean Squared Error) and RMSE (Real Mean Squared Error). The results show that the hybrid model with wavelet packet transform in the first phase more accurately forecasts than the model using stationary wavelet transform. In each of the hybrid schemes, the increase in resolution level, perhaps unexpectedly, brings no improved results.

This paper is organized as follows. The next section considers related work. The section 3 presents a review of theory behind wavelets and neural networks. The section 4 covers all aspects of the introduced hybrid model, including a detailed description of the procedure for its design. In the section 5, the experimental results are presented to demonstrate the effectiveness of the presented hybrid strategy. The final section proposes conclusions and recommendations for future research.

## 2. Related work

The exchange rates are inherently non-stationary, noisy and chaotic time series [2]. They are a combination of long and short memory processes imbedded in one complex signal, explaining why their prediction can present a true challenge [2], especially knowing that due to its congenital complexity, traditional statistical methods perform poorly in this field. All of this actually suggests that there is no complete information base on which we can forecast this key economic factor. Moreover, the general assumption made in these cases is that the historical data of one time series integrates all important features necessary for successful prediction. Despite this complicated scenario, our goal in this paper is to investigate the use of wavelets and artificial neural networks for the prediction of several exchange rates.

Over the last few decades, it has become obvious that linear models do not adequately represent nonlinear series, while wavelet analysis theory has emerged as a powerful tool in the mathematical analysis field [3]. Simply said, the wavelet transform produces a functional local decomposition of a signal in both the time and frequency domains and is not restrained by the assumption of stationarity [3]. Numerous publications describe the application of wavelets in the field of finance [4, 5]. The two transforms that we use in this paper are wavelet packet and stationary wavelet transform. Both offer the capability of capturing key features of an underlying process with a limited number of coefficients. The fundamental and novel contribution of this paper is to use these two processing techniques to decompose an exchange rate into a set of approximation and detail series which are fed into the neural networks in the model's next stage.

The traditional approaches to time series prediction, such as Box-Jenkins or ARIMA method, assume that the time series used for the prediction process are linear and stationary [6]. Therefore, these methods are obviously not a good tool for exchange rate prediction. On the other hand, during the past few decades the Artificial Neural

Networks (ANNs) have shown great applicability in time series prediction [7, 8]. Studies have compared the performance of neural networks to ARIMA [9], with all research agreeing that ANNs perform better than ARIMA models. Several unique features of ANNs make them an attractive forecasting tool: they are multivariate, nonparametric statistical methods that can map any nonlinear function without a priori assumption about the data, yet maintain desired accuracy [8]. Numerous articles illustrate the practical considerations of ANNs' applicability [10, 11].

By combining wavelet transform with artificial neural networks we get a new kind of modeling method with great prediction ability for high frequency financial data. With this synergy, we gain advantages from both of the methods - the multiscale analysis supplied by wavelet theory and powerful learning and training capability of the neural network. Amongst many studies that investigate the concept of mixing wavelets and neural networks, we refer to several in our research [12-14].

The occurrence of hybrid systems, namely the combination of neural networks with wavelet transform, has established itself as the next logical step in the improvement phase of the exchange rate forecast. Besides this combination, it is important to note that other innovative hybrid learning algorithms exist, such as the combination of neural networks with genetic algorithms and fuzzy logic [15]. Since it is clear that each method has its pros and cons, one must keep in mind that the weaknesses of one method can be supplemented by the strengths of the other, thus creating a single complementary technique that has better results than the two separate methods.

Genetic algorithm (GA) is an adaptive heuristic algorithm based on the analogy with biological evolution. This method is often used in combination with neural networks due to its convenience for parameter optimization and robustness – in other words, the algorithm is flexible and expands easily if necessary. GA units represent momentary approximations of problem resolution, while each unit is accompanied by so-called fitness function whose values give an insight in how “capable” the unit is. Following the selection, hybridization and mutation processes, every fitness function value of unit is compared and the lowest units are eliminated. Consequently, each new generation is of better quality until the formation of the most optimal result. There is a significant amount of literature available that demonstrate successful role of GA as optimization tool in combination with neural networks in forecasting the exchange rates [16, 17].

Fuzzy logic is the concept that does not clearly define affiliation of one element to the group. Instead, the affiliation is measured by a value between 0 and 1. This hybrid intelligent model combines fuzzy systems and neural networks, or in other words, the human-like perception feature of the fuzzy system is combined with the neural networks' learning and generalization characteristics. The advantage of these methods is the relatively simple use of the fuzzy logic and the consequent flexibility of the entire algorithm. The research that demonstrated the successful exchange rate forecast with the aid of this type of hybrid models is [18].

### **3. On the use of wavelets and neural networks**

This section explains the value of wavelets in time-series prediction together with neural networks.

### 3.1. The value of wavelet transform

One of the drawbacks of the Fourier analysis is that although it is possible to determine the frequencies present in the signal with this analysis, it is not possible to establish when they actually occur [19]. This premise is surpassed with the introduction of wavelet transform, representation of the signal whose root is comprised of wavelets (or mother wavelets or analyzing wavelets). Wavelet transform, through the mechanism of mother wavelets translation, offers precise information about time and frequency resolution [3].

Discrete Wavelet Transform (DWT) is one of the most convenient tools for signal analysis [3]. With its help, we can present the signal with a limited number of coefficients that capture information at different frequencies at distinct time moments. One of the variations of the discrete wavelet transform is Wavelet Packet Transform (WPT), where both approximation and detail coefficients are decomposed ( $2^n$  sets of coefficients are produced, unlike  $n+1$  in the case of a DWT). With this kind of signal presentation, the most complex and detail signal presentation is gained. Figure 1 shows level 2 wavelet packet decomposition.

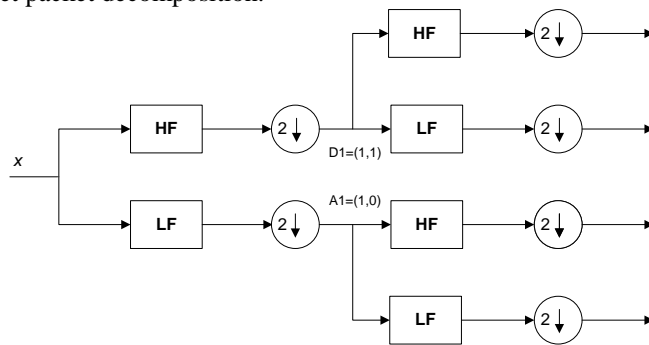


Fig. 1. Wavelet packet decomposition of level two.

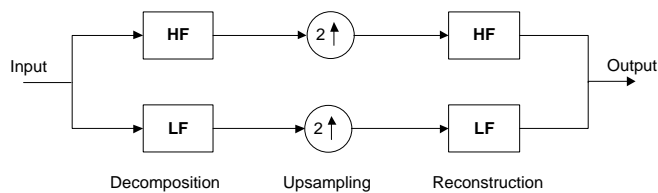


Fig. 2. Filter bank scheme for stationary wavelet transform.

A significant problem with DWT is its shift variability and aliasing. Additionally, the disadvantage of this algorithm is the fact that the length of the coefficient series decreases while the iteration index increases due to the use of the decimators [20]. Shift variance problem led to the idea of eliminating decimation operation and the formation of Stationary Wavelet Transform (SWT) (Figure 2). The result of such an idea is the shift-invariable transform, also described in literature as *à trous* or redundant wavelet

transform [13]. Stationary wavelet transform can be presented through successive convolutions with the discrete low-pass filter  $h$ :

$$c_{i+1}(k) = \sum_{l=-\infty}^{+\infty} h(l)c_i(k + 2^i l) \quad (1)$$

where the zero scale is the original input signal  $c_0(t)=x(t)$ .  $2^i l$  presents the increase in distances between sampled points and that is why this transform also goes by name *à trous*, meaning "with holes".

Wavelet coefficients are obtained by taking the difference between successive smoothed versions of the signal:

$$w_i(k) = c_{i-1}(k) - c_i(k). \quad (2)$$

Having in mind all previously stated, the original signal can be presented as:

$$x(t) = c_p(t) + \sum_{i=1}^p w_i(t). \quad (3)$$

As for the wavelets used in previous studies, most signal processing researchers adopted *Daubechies* and *Haar* wavelets [21]. The *Haar* wavelet transform is a good choice for edge detection and localized jumps. It is a commonly used tool in the analysis of financial time series i.e. for capturing fluctuations between its neighboring observations. The *Haar* mother wavelet is defined as:

$$\Psi(x) = \begin{cases} 1, & 0 \leq x < \frac{1}{2} \\ -1, & \frac{1}{2} \leq x < 1 \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

$$\phi(x) = \begin{cases} 1, & 0 \leq x < 1 \\ 0, & \text{otherwise.} \end{cases} \quad (5)$$

As for the Daubechies wavelets, many transforms have been created [22], however one of the most commonly used in time series analysis is Db40.

With all previously specified signal processing techniques, the input time series can be analyzed at multiple time resolutions; the signal can be smoothed until the long-term trend is identified and the fluctuations around the trend can be investigated at multiple time scales. After the decomposition, the individual time series can give a detailed and more easily analyzed view of the inner underlying processes.

### 3.2. Artificial Neural Networks

Artificial Neural Networks (ANN) are a class of nonlinear models that can extract crucial parameters from complex high-dimensional time series and approximate any

nonlinear function with a high level of accuracy as a result [23]. They are capable of discovering the underlying pattern or auto-correlational structure in the time series even when an underlying law is unknown or hard to determine, making them a powerful forecasting tool in many different fields. Despite the fact that neural networks have been successfully implemented in prediction process on numerous occasions, designing a predictor for specific financial time series with neural network is a challenging and nontrivial task. In comparison with Box Jenkins ARIMA models and other regressive models, a larger number of factors play a role in the neural networks design.

One of the most popular and most successfully implemented neural network models is the feed forward multilayer network or MLP (*multi-layer perceptron*) [24]. This type of network consists of several layers that contain nodes (artificial neurons). The first layer is the input level and receives external information, while the last layer is the output level and produces model solutions. Hidden layers lie in between these two. All of the nodes in one layer are connected to the nodes in the adjacent layer by an interconnection strength called weights. These weights are set through a training algorithm, where the goal is to minimize the difference between the network's target and actual output.

The design of neural network is a difficult task concerning the number of factors that influence its performance. Although some studies [25] propose certain methods for neural network design, no study has been reported to analytically determine general architecture rules for successful neural network design. One of the most sensitive parameters are number of layers, number of neurons in each layer, activation function (function that produces an output based on input values entering the node) and learning algorithm (the way the weights are set). The number of input and output layers depends on the problem's nature (in most papers, the suggested value is one). As for the hidden layers (internal information processing layers), it has been pointed out that one hidden layer network is able to approximate most of the nonlinear functions [27]. The dimension of each layer, rather the number of neurons in each layer, is one of the most essential parameters for the network's proficiency and successful performance [25]. Increased training time and reduced generalization ability of the NN can be a result of too many hidden nodes. On the other hand, if it is too few, the network's ability to learn will be reduced. In most cases, because of the lack of a systematic approach to neural network design and established guidelines for it, trial and error approaches are suggested for most previously stated architecture parameters.

To summarize, neural networks are difficult to design, require high processing and training time, and give unstable results in many situations. However, if their architecture is correctly planned (which demands a significant level of invested time and resources), they present a powerful tool that can perform many demanding tasks that linear programs cannot and can therefore be successfully implemented in many applications.

### 3.3. Statistical feature extraction

Besides basic wavelet features, we implemented an additional seven statistical features, all of them commonly used in finance and economics, into our model in order to improve the prediction process. They are given in the Table 1, where  $x$  presents the time series and value of current observation,  $n$  presents the total number of observations,  $\sigma$

the standard deviation,  $t$  represents the moment of time between 1 and  $n$ , and  $p_1$  to  $p_n$  are probabilities of the signal.

**Table 1.** Statistical features and their definitions

Statistical feature	Definition
Mean	$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$
Mean Absolute Deviation	$MAD(x_1..x_n) = \frac{1}{n} \sum_{i=1}^n  x_i - \bar{x} $
Variance	$VAR(x_1..x_n) = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2$
Skewness	$SKEW(x_1..x_n) = \frac{1}{n} \sum_{i=1}^n \left[ \frac{x_i - \bar{x}}{\sigma} \right]^3$
Kurtosis	$KURT(x_i - x_n) = \left\{ \frac{1}{n} \sum_{i=1}^n \left[ \frac{x_i - \bar{x}}{\sigma} \right]^4 \right\} - 3$
Turning Points	$(x_{t+1} - x_t)(x_t - x_{t-1}) < 0$
Shannon entropy	$H(x) = -\sum_{i=1}^n p_i \log_2 p_i$

#### 4. Hybrid Modeling Strategy

The main concept behind the prediction method presented in this paper is to decompose the studied exchange rate, using two different wavelet transforms, into a range of frequency scales and to pull these individual components through separate neural networks, making an aggregation forecast in the final neural network. The entire process of this algorithm involves a series of steps, including: statistical feature extraction, preprocessing step, wavelet analysis, neural networks training and modeling, and final forecasting. We approach this issue by dividing the model into three separate stages:

First stage: an exchange rate is processed and subjected to the wavelet-based decomposition process in order to detect underlying processes (features) for further evaluation.

Second stage: all individual decomposed components are fed into a set of neural networks in order to capture valuable information.

Third stage: the outputs from the neural network, rather the predicted values of each component, along with a set of statistical features (calculated on the original time series) are fed into the final neural network after which the prediction is to be made.

The data preparation phase includes the statistical parameters calculation and normalization of the exchange rate. Each statistical feature for a specific sample is calculated based on 10 previous samples. The normalization process has to be done in

order to avoid the effect of oversized values on the model and to fasten the calculation. It is carried out in such manner that both the inputs and targets fall in the range  $[-1,1]$ .

The input data are decomposed into a certain number of sub-time series components with the help of wavelet decomposition. Our main considerations regarding which wavelet transform and mother wavelet we use are:

- The input time series is discretely sampled so there is dyadic relationship between resolution scales, leading to the use of a DWT, rather its variation of a WPT.
- The goal is to have an  $n$ -length resolution scale for every resolution level leading to a *trous* wavelet transform.
- Wavelet functions which respect the complex nature of the financial time series lead to the use of *Haar* and *Daubechies* wavelet functions.

The choice of optimal decomposition level, generally depending on the researcher's experience, empirical data, and time series nature, is one of the most important factors of the model's performance in the first stage. In this study, we use two and three decomposition levels for both wavelet packet and stationary wavelet transform (poor results are obtained when using decomposition levels greater than three). Thus, the selected exchange rates have been decomposed using two different transforms in the following way:

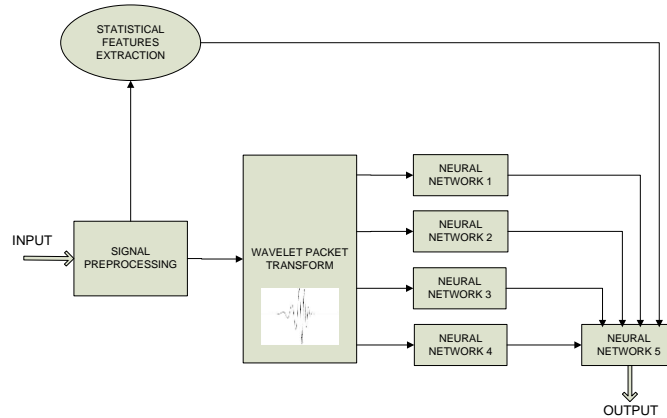
- Wavelet Packet Transformation with *Daubechies40* wavelet and decomposition levels of two and three.
- Stationary Wavelet Transform with *Haar* wavelet and decomposition levels of two and three.

Figures 3 and 4 show a representation of the models. For simplicity, both models have been considered only for a decomposition level of two.

The concept of using wavelets in time series analysis offers the advantage of separating the smooth part (approximation series) and the irregular and noisy (detail series) part of the signal, which are both more stable to handle and easier to predict due to the filtering effect of the transforms. As a result of this stage, the objective is adjusted in order to exploit these series as input signals to a set of neural networks in the following stage.

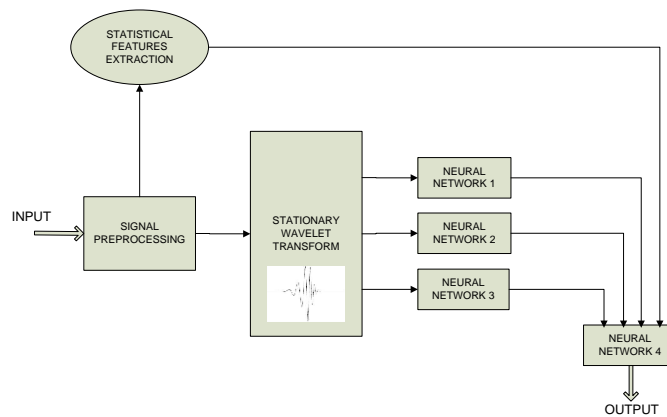
The results of the WP decomposition of level two are four sets of coefficients and as for the SWT of level two, the results are three sets of coefficients (one set representing approximation and two sets details). Analogous to that, the results of the WP decomposition of level three are eight sets coefficients, while there are four sets coefficients for the SWT of level two.





**Fig. 3.** Hybrid model with wavelet packet transform; the 2nd level of resolution.

The second stage of the model consists of a set of feed forward neural networks that uses lagged detail and smooth coefficients gained from decompositions in the previous step as input. One of the papers where a similar idea is presented is [4]. Separate neural networks are built for each decomposition level, meaning that a level 2 wavelet packet decomposition results in four neural networks, while level 2 stationary wavelet transform results in three neural networks.

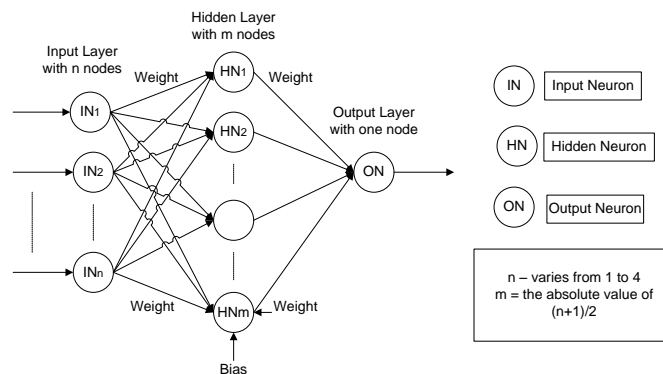


**Fig. 4.** Hybrid model with SWT decomposition; the 2nd level of resolution.

What can be noticed here is that we have a small amount of control over the complexity of the architecture in this stage. One possible way of handling this complexity is to test networks with different designs and compare them in order to choose the optimal one. Having this in mind, the problem can be tackled by varying a large number of design factors that influence the prediction result. This is the reason why we vary the number of input nodes and why we design the hidden layer as simply as possible. Not only does our research confirm that this is the best approach, but our

findings are also supported by literature which states that the simplest model is the least likely to overfit/underfit and the most likely to generalize well on the unseen data [27].

As for the design of each neural network, we apply the same architecture for each of them. In order to determine the optimal number of input nodes, we tested the ANNs with input layer that consists of 1, 2, 3 and 4 nodes. These input nodes process wavelet coefficients of the exchange rate gained in the first stage. Each neural network of the second stage takes the wavelet coefficients of the corresponding resolution level, as shown in Figures 3 and 4. We also noticed by a trial and error method that number of input nodes greater than four generates unstable output. The main reason can be in overtraining of the neural networks. The standard for determining the optimal number of input nodes is RMSE (Root Mean Square Error). Due to the fact that the NN with a single hidden layer can approximate any function with arbitrary precision and because input layer has fewer nodes, we design all NNs with a single-hidden layer. Having one output also contributes to this decision. For the number of nodes in the hidden layer, we apply a principle most often used in literature – Okam’s razor principle – where the number of hidden nodes equals half of the input and output nodes (we notice that increasing the number of hidden nodes or even adding more layers does not improve networks’ performance). Also having in mind that we want to predict a single value, we use one neuron in output layer. Figure 5 shows the MLP network used in the model’s second stage.



**Fig. 5.** An example of the feedforward multilayer neural network from the second stage. The inputs are samples obtained as output of the corresponding wavelet decomposition. The output of this neural network is forwarded to the corresponding input of neural network in the last stage of the presented model.

All series are split into two sets: training and testing. There are no specific rules for data division between these sets, and most researchers use a trial and error approach. For the training phase, we use a method known as the „sliding window” technique where the  $n$ -tuple input goes through the entire training set while a single output is used as the target value. Once trained, the networks are used for prediction.

As for the transfer functions, we use a linear one for the node in the output layer and a *tan sigmoid* function for the nodes in the hidden layer. This function is the most commonly used function in forecasting problems and pattern detection because it outperforms the alternatives when deviations are calculated from the average behavior [25]. All of the networks are trained using the *Scaled Conjugate Gradient* algorithm, a

supervised learning algorithm that shows linear convergence on the most of the problems [28] and provides faster learning.

In the final stage, the individual predictions along with statistical features that are calculated on the original time series are combined to generate an aggregate forecast. All of those values serve as inputs into the last neural network, where the final output is the one-step-ahead predicted sample. To design the last neural network, we use the same parameters as we did for the networks from the secondary stage of the model, with the only difference being that the number of input samples is fixed. Table 2 summarizes the architecture of the final network in the system, depending on the type of wavelet transform and level of resolution used in the second stage.

**Table 2.** The architecture details of the neural network in the last stage

Method	Resolution Level	Number of inputs	Number of Outputs	of NN architecture
Stationary Wavelet	2	10	1	10:5:1
Stationary Wavelet	3	11	1	11:6:1
Wavelet Packet	2	11	1	11:6:1
Wavelet Packet	3	15	1	15:8:1

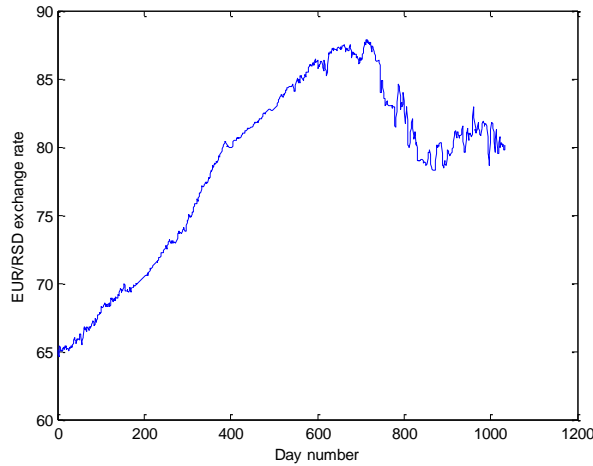
It should be pointed out, that we have designed a MATLAB program which implements the proposed model. The designed program takes a time series as input and gives predicted samples as output. It allows one to change models parameters such as: wavelet function, decomposition level, etc. The program also performs statistical analysis and calculates the performance of the model. It offers wide range of fine parameter tuning for the discussed model and presents a scalable and flexible solution for further research.

To conclude, based on the previously introduced model, the goal is to demonstrate that the one-step-ahead predictions for EUR/RSD exchange rate can be estimated with reasonable accuracy. The predictive power of these forecasts is compared by using a set of statistical parameters. In the next section, we will show that the model's ability to capture dynamical behavior differs with the wavelet resolution level, but not in a way that we expect. This leads to a discussion of our results according to the two types of wavelet transforms used.

## 5. Results

In this Section, we present results of the proposed hybrid model applied for prediction of three different exchange rates. The development of the hybrid model has been initiated in order to build an accurate predictor for domestic currency, Serbian Dinar, and its exchange rate to Euro, i.e. EUR/RSD exchange rate. Thus, the main focus here is to analyze prediction of EUR/RSD exchange rate in details. In order to test robustness and applicability of the proposed hybrid model, we also apply it for prediction of exchange rate of Hungarian Forint to Euro, EUR/HUF. Hungarian Forint has similar features as RSD, and we expect similar results. Finally, we also use the hybrid model for prediction

of exchange rate of Great Britain Pound to Euro, EUR/GBP. The GBP has different features compared to RSD, and obtained results are very interesting.



**Fig. 6.** EUR/RSD exchange rate from July 2003 till September 2007.

The input data used in the first case is the official exchange rate of Euro against the domestic exchange rate (Republic Serbia Dinar) between July 2003 and September 2007 (total length of 1024 samples). We believe that this period of exchange rate is the most suitable for testing because it contains various dynamic changes in exchange rate. The first 80% of it is used as sample data for the training phase; while the remaining 20% is used for evaluation of each neural network. We choose this kind of data division because for others we notice slightly irregular model behavior which can be explained with poor generalization in case of smaller training sets and overtraining effect in case of larger training sets. The graph of the EUR/RSD exchange rate is illustrated in Figure 6. For all tests and simulations, we use the special MATLAB program explained above.

As for the prediction performance, the hybrid wavelet neural model is evaluated by using four statistical parameters: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Square Error (MSE) and Real Mean Square Error (RMSE). These parameters are defined in the following manner:

$$MAE = \frac{1}{n} \sum_{t=1}^n |y_t - \hat{y}_t| \quad (6)$$

$$MAPE = \frac{1}{n} \sum_{t=1}^n \frac{|y_t - \hat{y}_t|}{y_t} * 100\% \quad (7)$$

$$MSE = \frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2 \quad (8)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2} \tag{9}$$

where  $y_t$  is the real value and  $\hat{y}_t$  is the predicted value. Although most of the above expressions are self-explanatory, it is useful to point out the following: MAE measures the deviations between the actual and predicted values, MAPE is the average absolute percentage error, MSE is the average of the squared errors between the predicted and real value, and RMSE presents how good a variance of the estimate is. Obviously, the closer that these values are to zero, the more accurate is the prediction performance.

Following the presented modeling strategy, the EUR/RSD exchange data is preprocessed and decomposed into two different resolution levels by WPT and SWT. The goal is to make underlying temporal processes of the original exchange rate more traceable and easier for further analysis. In the case of SWT, the *Haar* wavelet transform gives the best results, unlike in the case of WPT where one of the wavelets from the *Daubechies* family worked best. Afterwards, the new set of sub-time series is fed into a set of neural networks.

In this phase, we also tackle the denoising effect on the prediction performance. The goal is to check if the process of noise removal will improve the quality of the overall final forecast. We apply denoising in the first stage with the wavelet packet transform used. Because the process of noise removal is quite complex, we notice that the model performance is particularly sensitive to the threshold parameter and that it is very important to determine the correct threshold value and apply denoising to the detail coefficients [29,30]. The process of soft thresholding has been tested for various values from 0.01 to 0.06 with step of 0.005, and the best results are obtained for the threshold value of 0.02. After filtration, wavelet packet reconstruction is performed to obtain a denoised signal that serves as an input signal in the next phase. It can be noticed that the noise from the original time series is removed without the influence of sudden glitches which means that most of the original signal is preserved. This feature is one of the biggest advantages of the wavelet packet method.

**Table 3.** The prediction performance of each neural network in the second stage for the selected model based on WPT of level two (shown in Fig 3), measured in RMSE

	NN1	NN2	NN3	NN4
RMSE	0.0026	0.0667	0.1374	0.1356

In the next stage, we investigate the approach of forecasting each individual sub-time series in the set of neural networks. All networks are trained separately (by using corresponding wavelet coefficients), and the objective is to perform a one-day-ahead prediction for each time series. Depending on the wavelet transform used in the first stage, we train different number of neural networks in the second stage. For example, we train three neural networks in the case of SWT with two resolution levels and three outputs. In the case of WPT with two resolution levels, there are four neural networks, and this is the use case shown in Fig. 3. The number of neural networks in the second stage can be eight, if WPT with three resolution levels and eight outputs is used. The performance of forecasting of each individual sub-time series done by the set of neural networks in the model's second stage is measured by RMSE. These RMSE values for

selected model consisting of WPT of level two with four neural networks are given in Table 3.

The prediction results of each neural network are individually combined with statistical parameters, calculated from the original EUR/RSD time series, thus establishing the inputs for the final neural network. Figure 7 compares the real time series and the output of the last neural network (the simulated time series). This comparative visualization is given for both models and for each resolution level. There is a large degree of overlap in input and output values for each model and resolution level, with no major discrepancies.

The forecasting analysis is performed based upon the results from the one-step-ahead prediction of the presented hybrid model. We measure the performance metrics to investigate how well the model captures the underlying trend of the movement of the EUR/RSD exchange rate. Table 4 shows the performance metrics achieved by our model. We illustrate this performance in Figure 8 where the prediction of 100 samples for each model and resolution level is shown.

A quick look at the last figure cannot expose which model and resolution level shows the best modeling performance. However, according to Table 4 it can be seen that the model with the WPT scenario is superior to the model with the SWT used. This means that wavelet packet transform decomposes the signal in a more accurate and precise way and that the learning algorithm applied on those coefficients handles the underlying structures better than the model with a SWT scenario applied. When it comes to WPT and denoising applied, the results indicate that both models are running well, with slightly more accurate and stable results in the case where noise is removed.

Additionally, we observe that the ability of the model to capture dynamical behavior is changing with the resolution level used. Although one would think that for lower resolution levels, resulting in noise and irregular sub-time series, the model would show less accurate results, we actually get less accurate results with the higher resolution time series (in other words, when we use the more smooth series). This phenomenon can be summarized in the following way – the performance of the model deteriorates with an increase in the resolution level, which can be shown by the statistical parameters in Table 4. We believe that it happens because with the higher resolution level, we extract detail coefficients which are mainly short term noise. This information is not stable and convenient as input to the model's second stage, resulting in disruptive effect on model's overall performance. In both cases, the optimal decomposition level is two. An increase of the resolution level greater than three yields poor results and a meaningless prediction as a result.

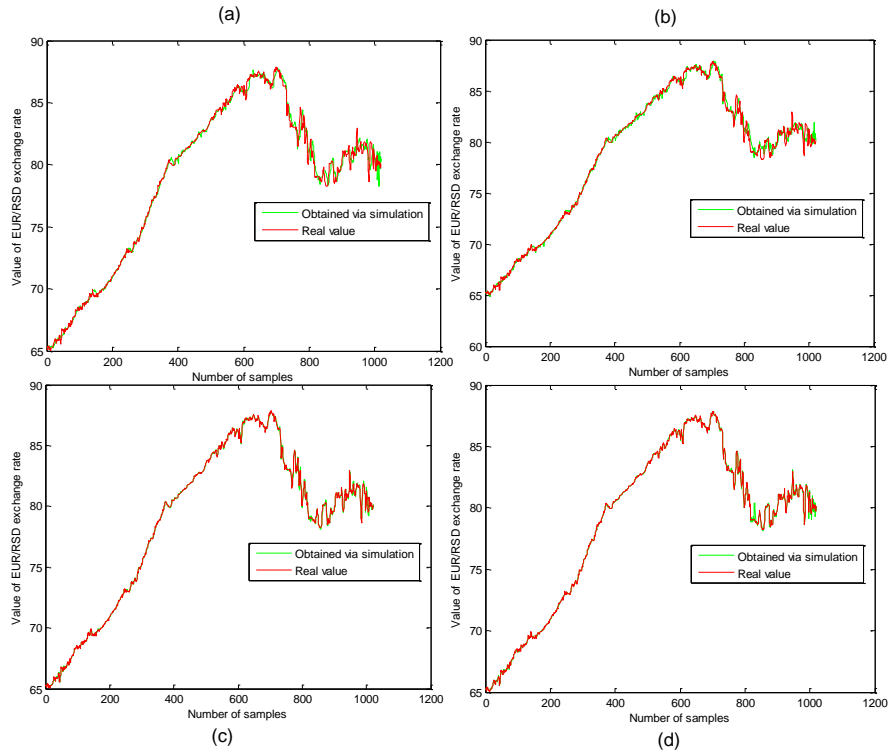
Evident from the preceding is that we here analyze an exchange rate with frequent, high jumps and peaks, which can corrupt the prediction process to a large degree. The model tests the EUR/RSD exchange rate over a very particular time period, during which the Republic of Serbia experienced a difficult financial crisis in correlation with a generally volatile global economy (our intention is to test the model on the most unstable part of exchange rate; we expect that by taking the EUR/RSD exchange rate from any other historical period the prediction process would obviously be easier with more accurate results). Due to this volatility, we believe that the trade market itself is not valid enough and that the historic data cannot depict all of the information required. This explains why we have boosted the model's accuracy with the introduction of statistic

features in the final stage. Having all of the preceding in mind, the model largely manages to predict the one day value of the EUR/RSD exchange rate.

**Table 4.** Performance metrics for EUR/RSD exchange rate depending on parameters used in model's stages

Wavelet transform	Level of resolution	Wavelet	Number of NNs in the second stage	Number of inputs in the NN in the third stage	Number of samples predicted	MAE	MAPE	MSE	RMSE
SWT	2	Haar	3	10	100	0.01080	0.00013	0.02246	0.01690
SWT	3	Haar	4	11	100	0.01094	0.00014	0.02307	0.01713
WPT	2	Db40	4	11	100	0.00486	0.00006	0.00450	0.00756
WPT	3	Db40	8	15	100	0.00597	0.00008	0.00746	0.00974

In the second case, we use the official exchange rate of Euro against the Hungarian Forint, EUR/HUF exchange rate, between July 2003 and September 2007 as the input data (total length of 1024 samples). We use the same period of EUR/HUF exchange rate because we believe that there are similar dynamic changes and economic factors with similar influence to HUF. According to common economical evaluation, HUF and RSD are very similar currencies in the economical sense. The first 80% of it is used as sample data for the training phase; while the remaining 20% is used for evaluation of each neural network. For all tests and simulations, we use the special MATLAB program explained above, following exactly the same procedure with all parameters. The prediction results of each neural network are individually combined with statistical parameters, calculated from the original EUR/HUF time series, thus establishing the inputs for the final neural network. Figure 9 compares the real time series and the output of the last neural network (the simulated time series) for different wavelet transforms and resolution levels. We measure the performance of the forecasting model using statistical metrics to investigate how well the model deals with movement of the EUR/HUF exchange rate. Table 5 shows the performance metrics achieved by our model in this case. According to Table 5, it can be seen that the model with the WPT scenario is superior to the model with the SWT used. This result is expected due to the very similar features of HUF to RSD, and all other conclusions are same as in the case of EUR/RSD. However, the statistical metrics of Table 5 show that the proposed hybrid model achieves approximately two times better accuracy in the case of EUR/HUF exchange rate. We believe that this is a consequence of more stable exchange rate. Compared to EUR/RSD, EUR/HUF maintains its mean value in a given time period although short time variations are present.

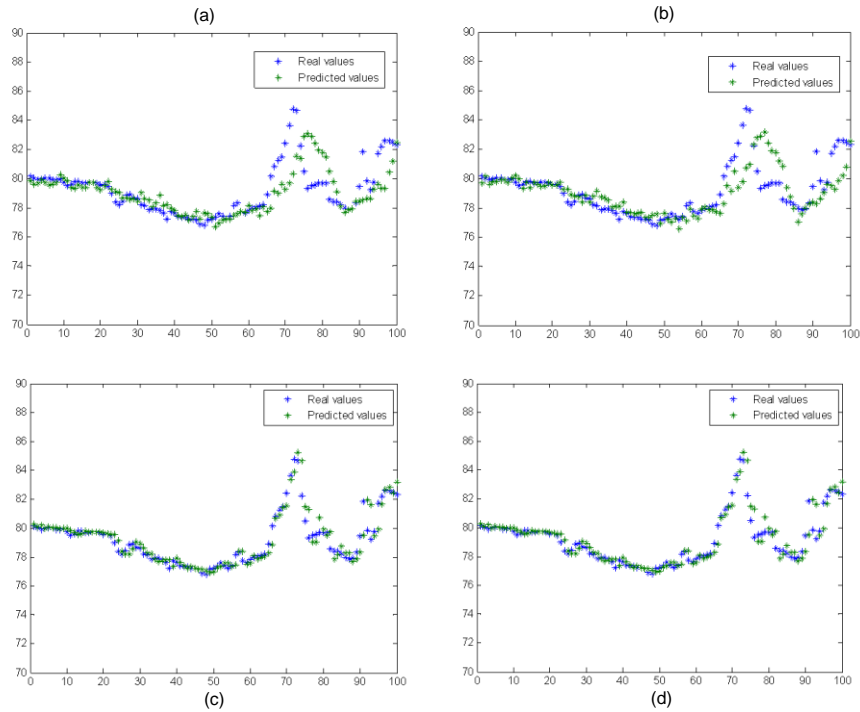


**Fig. 7.** Visual comparison of real and simulated signal in the EUR/RSD case (a) Stationary wavelet transform used, 2 resolution levels (b) Stationary wavelet transform used, 3 resolution levels (c) Wavelet packet transform used, 2 resolution levels (d) Wavelet packet transform used, 3 resolution levels.

**Table 5.** Performance metrics for EUR/HUF exchange rate depending on parameters used in model's stages

Wavelet transform	Level of resolution	Wavelet	Number of NNs in the second stage	Number of inputs in the third stage	Number of NN in samples predicted	MAE	MAPE	MSE	RMSE
SWT	2	Haar	3	10	100	0.00523	0.00021	0.01291	0.00721
SWT	3	Haar	4	11	100	0.02692	0.00011	0.35416	0.03791
WPT	2	Db40	4	11	100	0.00358	0.00001	0.00537	0.00465
WPT	3	Db40	8	15	100	0.00585	0.00002	0.01691	0.00826





**Fig. 8.** Comparative view of 100 real and predicted samples (a) Stationary wavelet transform used, 2 resolution levels (b) Stationary wavelet transform used, 3 resolution levels (c) Wavelet packet transform used, 2 resolution levels (d) Wavelet packet transform used, 3 resolution levels.

Finally, we test the proposed hybrid model in the case of currency that has very different features compared to HUF and RSD. In the third case, we use the official exchange rate of Euro against the Great Britain Pound, EUR/GBP exchange rate, between July 2003 and September 2007 as the input data (total length of 1024 samples). We use the same period of EUR/GBP exchange rate in order to obtain fair comparison. The first 80% of it is used as sample data for the training phase; while the remaining 20% is used for evaluation of each neural network. For all tests and simulations, we use the special MATLAB program explained above, following exactly the same procedure with all parameters. The prediction results of each neural network are individually combined with statistical parameters, calculated from the original EUR/GBP time series, thus establishing the inputs for the final neural network. Figure 10 compares the real time series and the output of the last neural network (the simulated time series) for different wavelet transforms and resolution levels. We measure the performance of the forecasting model using statistical metrics to investigate how well the model deals with movement of the EUR/GBP exchange rate. Table 6 shows the performance metrics achieved by our model in the EUR/GBP case. According to Table 6, it can be seen that the model with the WPT scenario is superior to the model with the SWT used. In fact, SWT cannot be used for prediction of GBP at all, because results are not accurate. Furthermore, the hybrid model with WPT gives superior accuracy in prediction of EUR/GBP compared to other currencies. The statistical metrics of Table 6 show that the

proposed hybrid model with WPT achieves approximately two times better accuracy in the case of EUR/GBP compared to EUR/HUF, and four times better compared to EUR/RSD exchange rate. The better accuracy is a consequence of more stable currency with smaller short time variations.

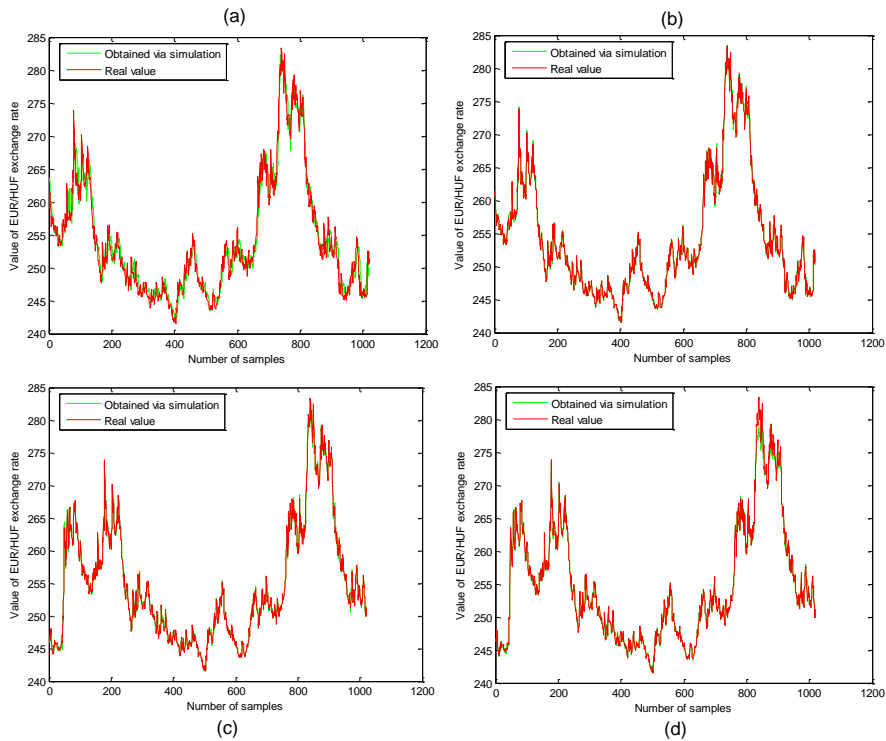
**Table 6.** Performance metrics for EUR/GBP exchange rate depending on parameters used in model's stages

Wavelet transform	Level of resolution	Wavelet	Number of NNs in the second stage	Number of inputs in the third stage	Number of samples in predicted	MAE	MAPE	MSE	RMSE
SWT	2	Haar	3	10	100	0.03851	0.05695	0.00141	0.04574
SWT	3	Haar	4	11	100	0.03821	0.05636	0.00137	0.04499
WPT	2	Db40	4	11	100	0.00177	0.0018	$3.9 \cdot 10^{-6}$	0.00236
WPT	3	Db40	8	15	100	0.00309	0.0021	$1.2 \cdot 10^{-5}$	0.00428

## 6. Conclusions and further research

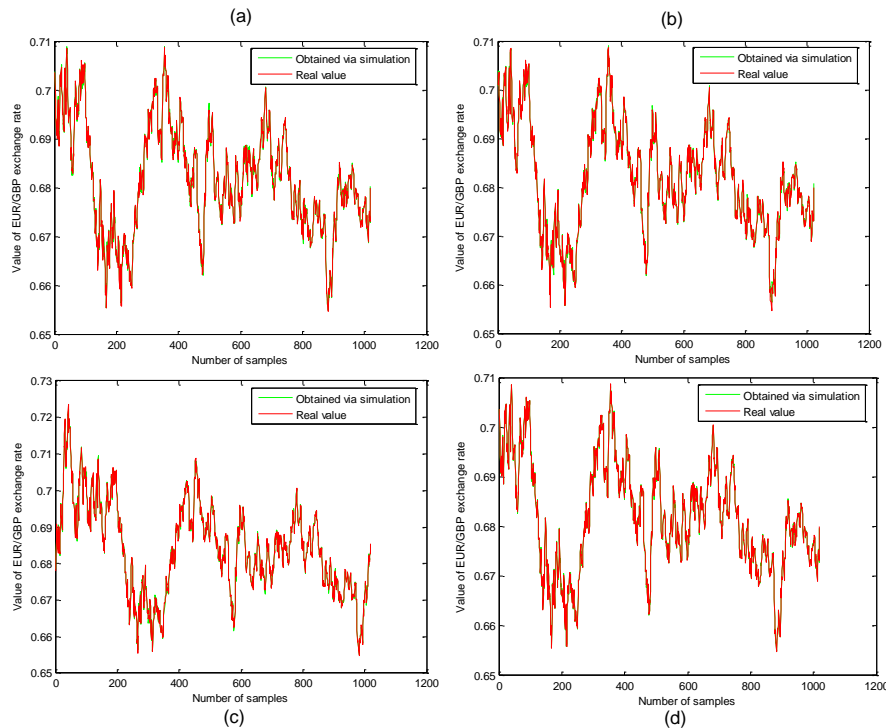
In this paper, we have analyzed prediction strategy combining two different wavelet transforms, neural networks and statistical features for prediction of three different exchange rates. According to our findings, both models present promising candidates for EUR/RSD exchange rate prediction. However, the proposed model with WPT scenario applied shows better performance in the case of EUR/HUF and especially EUR/GBP. Furthermore, the proposed hybrid model with WPT works better for more stable currencies, and achieves the highest accuracy in the case of EUR/GBP. It seems that the time series decomposed with WPT has a larger capacity to capture global behavior and thus offers richer information that is used in the second stage for the purpose of training, modeling and forecasting. Furthermore, the proposed system is more suitable for currencies that maintain their mean value.

Further results show that the noise can slightly reduce the generalization capability of a model. Another conclusion we have drawn is opposite to our expectations – increasing the resolution does not improve the system performance, indicating that a prediction may not necessarily be more accurate if the signal decomposition level is higher. We also believe that the presented algorithm can work for any time period and that its major contribution is actually its applicability for various nature of exchange rate. Of course, in every case the model has to be trained with different data sets, which having in mind the user-friendly side of the implemented software, is an easy task to do.



**Fig. 9.** Visual comparison of real and simulated signal in the EUR/HUF case: (a) Stationary wavelet transform used, 2 resolution levels (b) Stationary wavelet transform used, 3 resolution levels (c) Wavelet packet transform used, 2 resolution levels (d) Wavelet packet transform used, 3 resolution levels.

In the past years, numerous hybrid models have been investigated, and they all show significant promise. Future research could possibly study the predictive power for long term forecasts of the same model or the utilization of other outside imported economic indicators. Also having in mind that the predictive power of wavelet neural hybrid model is highly sensitive to a large number of parameters, we had to do tedious experiments and trial-and-error procedures in order to obtain valid results. These major weaknesses can perhaps be avoided by changing the selection of appropriate number of hidden nodes, training times and lags, and determining and setting systematic rules for these tricky tasks.



**Fig. 10.** Visual comparison of real and simulated signal in the EUR/GBP case: (a) Stationary wavelet transform used, 2 resolution levels (b) Stationary wavelet transform used, 3 resolution levels (c) Wavelet packet transform used, 2 resolution levels (d) Wavelet packet transform used, 3 resolution levels.

**Acknowledgements.** This paper is partly a result of the technology development project funded by the Ministry of Education and Science of the Republic of Serbia entitled "Performance optimization of energy-efficient computing and communication systems (TR 32023)".

## References

1. Kamruzzaman, J., Sarker, R.: ANN-Based Forecasting of Foreign Currency Exchange Rates. *Neural Information Processing - Letters and Reviews*, Vol. 3, No. 2, 49-58. (2004)
2. Yu, L., Wang, S., Huang, W., Keung Lai, K.: Are foreign exchange rates predictable? *Scientific Inquiry: A Journal of International Institute for General Systems Studies, Inc*, Vol. 8, No. 2, 207-228. (2007)
3. Holschneider, M.: *Wavelets: An Analysis Tool* (Oxford Mathematical Monographs). Oxford University Press. (1999)
4. Renaud, O., Stark, J.L., Murtagh, F.: Prediction based on a Multiscale Decomposition. *International Journal of Wavelets, Multiresolution and Information Processing*, Vol. 1, No. 2, 217-232. (2003)
5. Gencay, R., Selcuk, F., Whitcher, B.: *An Introduction to Wavelets and Other Filtering Methods in Finance and Economics*. San Diego: Academic Press Ltd. (2002)

6. Box, G. E. P., Jenkins, G. M., Reinsel, G.C.: Time Series Analysis: Forecasting and Control. the 4<sup>th</sup> ed. John Wiley & Sons. (2008)
7. Yao, J., Li, Y., Tan, C.L.: Option price forecasting using neural networks. OMEGA: International Journal of Management Science, Vol. 28, No.4, 455-466. (2000)
8. Yao, J., Tan, C.L.: A case study on using neural networks to perform technical forecasting of forex. Neurocomputing, Vol. 34, 79-98. (2000)
9. Kamruzzaman, J., Sarker, R.: Comparing ANN based models with ARIMA for prediction of forex rates. ASOR Bulletin, Vol. 22, 2-11. (2003)
10. Gomez-Gil, J., Ramirez-Cortes, J., Hernandez, S., Alarcon-Aquino, V.: A Neural Network Scheme for Long-Term Forecasting of Chaotic Time Series. Neural Processing Letters. Vol. 33, No. 3, 215-233. (2011)
11. Pacelli, V., Bevilacqua, V., Azzollini, M.: An Artificial Neural Network Model to Forecast Exchange Rates. Journal of Intelligent Learning Systems and Applications, Vol. 3, No. 2, 57-69. (2008)
12. Minu, K.K., Lineesh, M.C., Jessy John, C.: Wavelet Neural Networks for Nonlinear Time Series Analysis. Applied Mathematical Sciences, Vol. 4, No. 50, 2485-2595. (2010)
13. Murtagh, F., Starck, J.L., Renaud, O.: On Neuro-wavelet Modeling. Decision Support Systems, Vol. 37, No. 4, 475-484. (2004)
14. Fan, S., Ji, T., Gordon, W., Rickard, B.: Forecasting Baltic Dirty Tanker Index by Applying Wavelet Neural Networks. Journal of Transportation Technologies, Vol.3, No.1, 68-87. (2013)
15. Rajasekaran, S., Vijayalakshmi Pai, G.A.: Neural networks, fuzzy logic and genetic algorithm: Synthesis and Applications, PHI Learning Private Limited, New Delhi. (2011)
16. El Shazly, M.R., El Shazly, H. E.: Forecasting currency prices using a genetically evolved neural network architecture. International Review of Financial Analysis, Vol.8, No.1, 67-82. (1999)
17. Nag, A.K., Mitra, A.: Forecasting daily foreign exchange rates using genetically optimized neural networks, Journal of Forecasting, Vol. 21, 501-511. (2002)
18. Kodogiannis, V.S., Lolis, A.: Forecasting exchange rates using neural network and fuzzy system based techniques, Neural Computing and Applications, Vol. 11, 90-102. (2002)
19. Ortega, L., Khashanah, K.: A Neuro-wavelet Model for the Short-Term Forecasting of High-Frequency Time Series of Stock Returns. Journal of Forecasting, Vol.3, No. 2, 134-146. (2013)
20. Stojlojescu, C.: A Wavelets Based Approach for Time Series Mining, Ph.D. Thesis. Politehnica University of Timisoara and TELECOM Bretagne, Brest. (2011)
21. Elfouly, F., Mahmoud, M., Dessouky, M., Deyab, S.: Comparison between Haar and Daubechies Wavelet Transformations on FPGA Technology. International Journal of Computer, Information & Systems Science, Vol. 2, No.1, 395-400. (2008)
22. Vonesch, C., Blu, T., Unser, M.: Generalized Daubechies Wavelet Families. IEEE Transactions on Signal Processing, Vol. 55, No. 9, 4415-4429. (2007)
23. Hippert, H.S., Pedreira, C.E., Souza, R.C.: Neural Networks for Short-Term Load Forecasting: A Review and Evaluation. IEEE Transactions on Power Systems, Vol. 16, No. 1, 44-54. (2001)
24. Zhang, G.P., Qi, M. Neural network forecasting for seasonal and trend time series. European Journal of Operational Research, Vol. 160, No. 2, 501-514. (2005)
25. Kaastr, I., Boyd, M.: Designing a neural network for forecasting financial and economic time series. Neurocomputing, Vol. 10, No. 3, 215-236. (1996)
26. Incze, A., Ileana, I., Rotar, C.: The optimization of feed forward neural networks structure using genetic algorithms. In Proceedings of the International Conference on Theory and Applications of Mathematics and Informatics – ICTAMI 2004, Thessaloniki, Greece, 223-234. (2004)

27. Kamruzzaman, J., Begg, R., Sarker, R.: Artificial Neural Networks in Finance and Manufacturing. Idea Group Inc (IGI). (2006)
28. Garcia, J.O., Reyes Garcia, C.A.: Detecting pathologies from infant cry applying scaled conjugate gradient neural networks. In Proceedings of the European Symposium on Artificial Neural Networks, Bruges (Belgium), 349-354. (2003)
29. Bozic, J., Babic, D.: Predicting the EUR/RSD exchange rate using wavelets and neural network. In Proceedings of International Conference on Applied Internet and Information Technologies, Zrenjanin, Serbia, 108-112. (2013)
30. Bozic, J., Vukotic, S., Babic, D. Prediction of the RSD exchange rate by using wavelets and neural networks. In Proceedings of the 19<sup>th</sup> Telecommunications Forum TELFOR, Belgrade, Serbia, 703-706. (2011)

**Jovana Božić** received the B.Sc. and M.Sc. in electrical engineering from the School of Electrical Engineering, University of Belgrade, Serbia. She is currently a PhD candidate at the School of Computing, Department of Signal processing in telecommunications, of the Union University. Her main research interests are in the areas of time series prediction, artificial neural networks and wavelet-based signal processing.

**Dorđe Babić** received Diploma Engineer degree from the School of Electrical Engineering, Belgrade, Serbia and Doctor of Technology (PhD) degree from the Tampere University of Technology, Finland. From 1999 to 2004, he worked as a researcher at Telecommunication Laboratory, Tampere University of Technology. Since 2007, he has been employed as an associate professor for telecommunications at School of Computing, Belgrade, Serbia. During 2008, he received DAAD research grant for three months research visit to Ruhr-Universität Bochum, Germany. His general research area includes Multirate Signal Processing in Software Radio Receivers, and Software Radio Concept.

*Received: July 28, 2014; Accepted: February 27, 2015*