

Forecasting Cryptocurrency Time Series Using Fuzzy Transform, Fourier Transform and Fuzzy Inference System

Srdan Jelinek^a and Ana Poledica^b and Bratislav Petrović^b and Pavle Milošević^{cb}

^aFaculty of Organizational Sciences, University of Belgrade, srdjanjelinek@gmail.com

^bDepartment of Systems Theory and Control, Faculty of Organizational Sciences, University of Belgrade, {ana.poledica, bratislav.petrovic, pavle.milosevic}@fon.bg.ac.rs

Abstract

In this paper we propose a new approach for forecasting the cryptocurrency time series, which combines the fuzzy transform and the fuzzy inference system. We also test whether fuzzy transform yields better results forecasting results in comparison to Fourier transform. Finally, we will investigate whether fuzzy rules used in fuzzy inference system can successfully capture high and low volatility moments in the time series, as well as high correlations between the three different cryptocurrencies.

Keywords: Fuzzy Transform, Fourier Transform, Fuzzy Inference System, Cryptocurrency, Time Series, Forecasting

1 Introduction

Cryptocurrencies have been a new research topic in the financial world, especially in the stock markets during 2018. Although being present on the market for several years, popularity of cryptocurrencies raised and fell immensely in the last year due to overall hype and the fact that they have not been subject to regulatory measures yet. Given the increased trading volume, forecasting cryptocurrency time series is a hot topic for both practitioners and academic researches. However, methods used for forecasting are still mostly traditional statistical ones, which are used for any other time series, such as regression and some econometric methods such as ARIMA [1-3]. Soft computing methods [4] are also a popular choice for the time series forecasting, with neural network being one of the most popular tools, e. g. [5-7].

Besides neural networks, another popular soft computing tools which is used for time series prediction is Fuzzy Inference System (FIS), e.g. [8, 9]. To the best of our knowledge, there were no attempts to use FIS for forecasting cryptocurrency time series.

On the other hand, Fourier transform is successfully being used in many hedge funds and investment banks

for forecasting the prices of financial derivatives, usually options [10, 11]. The algorithm is to take the discrete Fourier transform (DFT) of the specific derivative or asset and then take the inverse DFT (IDFT) to get the future price. It also has many practical applications in digital signal or image processing, as well as solving partial differential equations or performing other operations such as convolutions [12]. Similar to FIS, there is no evidence for applying Fourier transform on cryptocurrency time series in order to obtain forecasts for the future periods.

The fuzzy (F) transform is a widely known tool for fuzzy modelling [13, 14], with most applications in time series processing and forecasting [15-17], image processing, compression, reconstruction [18-20], and in solving differential equations [21].

In this paper, we first want to test how F and Fourier transforms will perform when forecasting high-frequency and high-volatile data, such as cryptocurrency time series. The idea is to apply FIS together with F and Fourier transforms in order to capture the most volatile periods and correlations between the cryptocurrencies using the fuzzy rules. The reason for using FIS is that cryptocurrency is a relatively new asset on the financial market, and as such, many of its price drivers are still unknown. Therefore, our attempt is to model this vagueness using fuzzy approach. More about using fuzzy logic for modelling of vagueness is presented in [22]. To the best of our knowledge, neither F nor Fourier transforms have been used together with FIS in forecasting cryptocurrency time series. Further comparisons with standard forecasting techniques such as ARIMA, neural networks, etc. are also relevant to consider in future research.

The time series used for testing are the BitCoin, Ethereum and LiteCoin cryptocurrencies. As later explained in Section 3, cryptocurrencies have high correlations with each other, meaning that if one of the currencies has significant up or down trend, the others are most likely to follow the same trend in the near future. As said above, we try to model this behavior using fuzzy rules in FIS. Using of fuzzy rules in modelling is described in [23-25].

Finally, our last hypothesis is that technical analysis should be more successful when applied on cryptocurrencies because there is less insider information available due to nature of cryptocurrencies. Therefore, the number of asymmetrical information is reduced, meaning that technical analysis should be more successful in terms of prediction rate when applied for cryptocurrencies compared to regular financial assets.

The paper is organized in the following manner. A more detailed overview and summary of techniques used in this paper, F and Fourier transforms, as well as FIS is given in Section 2. The data used for testing are presented in Section 3. In Section 4 we propose an approach used for forecasting the time series. Results are presented in Section 5. Finally, a summary of the proposed approach, results and a future work is presented in Section 6.

2 Theoretical Background

2.1 Fuzzy Transform

F -transform is a widely used tool for fuzzy modelling. It converts a continuous or discrete function defined on a real $[a, b]$ domain into n -dimensional vector by establishing a mapping from a set of continuous functions defined on a domain of real numbers to the set of n -dimensional real vectors. The inverse F -transform maps the vector back to the space of continuous or discrete functions.

F -transform is computed by using n uniformly spaced membership functions defined on $[a, b]$ segment as weight ponders for a given function is to be mapped into vector space. Membership (or also called basic) functions $(A_i, i = 1, \dots, n)$ defined on their respective and equally spaced fuzzy sets represent the degree of truth in fuzzy logic, and are therefore covering the $[0, 1]$ range. They are mathematically defined and described in more detail in [26, 27]. A formal representation of the triangular membership functions is given in the Equation 1:

$$\begin{aligned} A_1(x) &= \begin{cases} 1 - \frac{x - x_1}{h_1}, & x \in [x_1, x_2] \\ 0, & \text{otherwise} \end{cases} \\ A_k(x) &= \begin{cases} \frac{x - x_{k-1}}{h_{k-1}}, & x \in [x_{k-1}, x_k] \\ 1 - \frac{x - x_k}{h_k}, & x \in [x_k, x_{k+1}] \\ 0, & \text{otherwise} \end{cases} \\ A_n(x) &= \begin{cases} 1 - \frac{x - x_{n-1}}{h_{n-1}}, & x \in [x_{n-1}, x_n] \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (1)$$

where $k = 2, \dots, n-1$ and $h_k = x_{k+1} - x_k$.

If the fuzzy segment $[a, b]$ is given by n basic functions, and if f is an arbitrary real continuous function, then direct continuous F -transform of f with respect to the given fuzzy interval is defined as the vector of n real numbers $[F_1, \dots, F_n]$, where i^{th} component of the F -transform of f , $[F_i]$, is defined as:

$$F_i = \frac{\int_a^b f(x) A_i(x) dx}{\int_a^b A_i(x) dx}, i = 1, \dots, n \quad (2)$$

If the f is a discrete function, and therefore defined on a set of T points, then direct discrete F -transform with respect to the given fuzzy interval is defined as the vector of n real numbers $[F_1, \dots, F_n]$, where i^{th} component $[F_i]$ of the F -transform of f , is defined as:

$$F_i = \frac{\sum_{t=1}^T f(x_t) A_i(x_t)}{\sum_{t=1}^T A_i(x_t)}, i = 1, \dots, n \quad (3)$$

Finally, an inverse F -transform of f is given as:

$$f_{F,n}(x) = \sum_{i=1}^n F_i A_i(x) \quad (4)$$

2.2 Fast Fourier Transform

Fourier analysis converts the function from its original domain, which is usually time or space, to a frequency domain by approximating the function with sums or basic trigonometric functions. The process of going back and rebuilding a function from the frequency domain is known as Fourier synthesis.

The DFT is the form of the Fourier analysis which operates on the equally-spaced samples of discrete data with interval usually being in units of time. It converts a signal in the time domain into its counterpart in the frequency domain and is defined as in Equation 5:

$$F_n = \sum_{k=0}^{N-1} x_k e^{\frac{-2i\pi nk}{N}} \quad (5)$$

An IDFT is then given by:

$$x_k = \frac{1}{N} \sum_{n=0}^{N-1} F_n e^{\frac{2i\pi nk}{N}} \quad (6)$$

Given the similarities between Equations 5 and 6, IDFT can also be computed using DFT [25].

The process of computing DFT from the time series, or its inverse, is called fast Fourier transform (FFT). The most common FFT algorithm is the Cooley-Tukey algorithm, which reduces the overall runtime to $O(N \log(N))$ when performed recursively, by breaking up a transform of length N into two transforms of length $N/2$ using the identity:

$$\sum_{n=0}^{N-1} a_n e^{\frac{-2i\pi nk}{N}} = \sum_{n=0}^{N/2-1} a_{2n} e^{\frac{-2i\pi(2n)k}{N}} + \sum_{n=0}^{N/2-1} a_{2n+1} e^{\frac{-2i\pi(2n+1)k}{N}} \quad (7)$$

2.3 Fuzzy Inference System

FIS is used to simulate the decision-making process using fuzzy logic. The system consists of multiple inputs, rules and outputs. Each input and output is modelled across the fuzzy space $[a, b]$ using the set of membership functions, where a and b can be different for each input/output. As described in Section 2.2, membership function of a fuzzy set A , maps each element to a value between 0 and 1. The value is called the degree of membership and it shows the level of membership of the element in $[a, b]$ segment to the fuzzy set A . Membership function is usually a simple function, triangular, trapezoidal or Gaussian, but it can also be custom-modelled. In most cases, using more complex functions does not yield higher precision as one is working in fuzzy universe.

Fuzzy rules are used to describe the relationship between inputs and outputs, and are modelled in an IF-THEN manner. IF conditions can also be combined to create more complex rules (i.e. IF $a=A$ AND $b=B$ THEN $c=C$).

The process of obtaining outputs using FIS can be split into three subprocesses. First subprocess is fuzzification, where crisp input values are transformed into fuzzy sets using membership functions. Second subprocess is application of fuzzy rules to the fuzzified inputs (this process is similar to human reasoning). Third subprocess is defuzzification, where the fuzzy set obtained after applying rules is transformed back into crisp value(s), which is the output of the FIS. We applied Mamdani's method of fuzzification / defuzzification, as it is the most common method used in FIS due to its simple structure.

3 Data

Cryptocurrency daily time series used in this paper are obtained from the open-source database of crypto data - cryptodatadownload.com. The cryptocurrencies used

are BitCoin, Ethereum and LiteCoin. The time range is from 7/8/2015 to 24/12/2018, resulting in total of 1236 daily prices and daily trade volumes for each cryptocurrency. The database contains *open*, *close*, *high* and *low* prices, and for our analysis, we used time series of *open* prices, as these are the prices which are used for the executing pending overnight transactions next morning. All three price time series are shown on Figures 1-3. LiteCoin price time series has very high correlation with both BitCoin and Ethereum time series, which is around 95%. This is the information we want to use in our FIS by adding rules which should represent that mutual correlation.

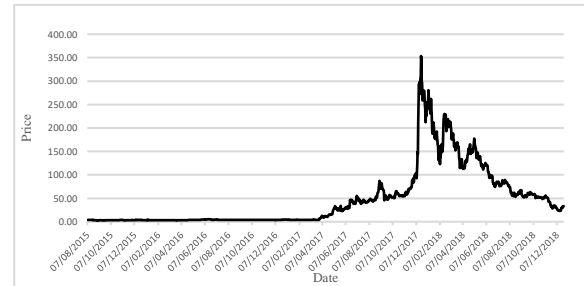


Figure 1: LiteCoin time series.

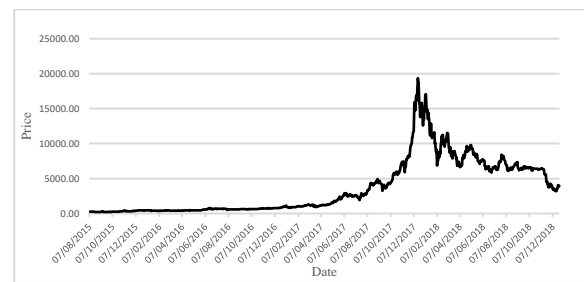


Figure 2: BitCoin time series.

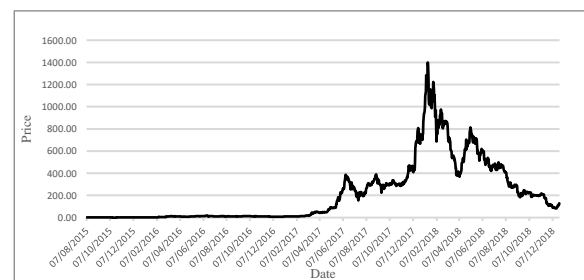


Figure 3: Ethereum time series.

4 Proposed approach

The whole process of forecasting cryptocurrency time-series consists of following steps:

1) Data preparation – calculation of daily relative price changes for each of the chosen cryptocurrencies, BitCoin, Ethereum and LiteCoin.

2) Prediction of the price on the $n+1^{\text{th}}$ day on the basis of n previous daily prices using F -transform and inverse F -transform.

- 3) Prediction of the price on the $n+1^{\text{th}}$ day on the basis of n previous daily prices using Fourier transform.
- 4) Creating FIS membership functions for inputs and output, as well as set of rules to be applied. Input values are trade volume and forecasted cryptocurrency prices obtained using F and Fourier transforms. Rules used are explained in Section 4.2.
- 5) Running FIS for two iterations, first one being with inputs obtained using F -transform, and the second one being with inputs obtained using Fourier transform.
- 6) Obtaining crisp output(s) using FIS and comparing results.

In the following subsections, we present forecasting procedures and FIS model in more detail.

4.1 Forecasting procedures

We used two techniques for forecasting the data, F and Fourier transform. In order to test the model performance under real settings, we chose the data from 2015 to 2018 that include both flat and volatile periods. Since no typical training for Fourier and F transform is required, we test the model using moving window throughout the whole dataset. Given the fast data flow for cryptocurrencies, we picked the time range of 6 days. In other words, we consider 6 points in time, and uses them to forecast the price for the 7th day. That results in total of 1230 forecasted prices for each of the three cryptocurrencies.

The idea of the forecasting procedure is similar to the one presented in [29], but the realization and techniques used in model (linear regression) and testing (moving window) are significantly different.

First, we defined 10 triangular membership functions $[A_1, \dots, A_{10}]$ on the $[1, 6]$ 6-day segment and constructed according to Equation 1. The value of h was empirically chosen to be 0.6. Then we calculated vector of 10 F -values $[F_1, \dots, F_{10}]$ using the Equation 3. Next, we added one more membership function (A_{11}) , as membership functions are equally distanced and identical. The (F_{11}) is then predicted using least squares regression on previous 10 F -values. We tested the order of the fitted polynomial from 1 to 5, but given that using higher order did not yield in significantly better predictions, we used the linear regression.

Finally, forecasted price f_7 is calculated based on 11 values of membership functions $[A_1, \dots, A_{11}]$ and a vector of corresponding F -values $[F_1, \dots, F_{11}]$ using Equation 4.

The second technique used for forecasting data is the FFT. The window size of 6 days is the same as for the

F -transform, and we are predicting the price for the seventh day.

First, we used the least squares regression to fit the polynomial through the 6 daily price points. Similar to the process applied for F -transform, we compared the results obtained using the different order of the polynomial, from 1 to 5, and again decided to use the linear regression as it yielded best results. Next, we calculated differences between the fitted polynomial line and the actual prices. Then, Fourier transform is applied on the differences. Next, a theoretical difference between the actual price on the 7th day is forecasted by taking the IFFT of the obtained time series and extrapolating it for one more day using Equation 5 and the process described in [28]. The last step is applying forecasted difference on the fitted polynomial curve for the 7th day.

By using linear regression instead of some non-linear predictor, we assume implicitly that prices in a 6-day segment define a trend and that it will be followed on the 7th day. This assumption could potentially hurt forecasting performance in highly volatile periods. However, bearing in mind that trends in observed time series are mostly strong and long-term, the choice of linear regression as a predictor seems to be justified.

4.2 FIS structure

FIS structure has 4 inputs, a set of rules and 1 output. First 3 inputs are the relative price changes obtained from forecasted price time series for each of the 3 cryptocurrencies by using either F or Fourier transforms. For each of those 3 inputs a set of 4 membership functions is created. The membership functions are shown in Figure 5, where there are 2 membership functions which depict *strong* relative changes in price and are modelled using the normal distribution with means of +10% and -10%, and 2 membership functions which depict *weak* relative changes in price and are modelled using Normal distribution with means of +2% and -2%.

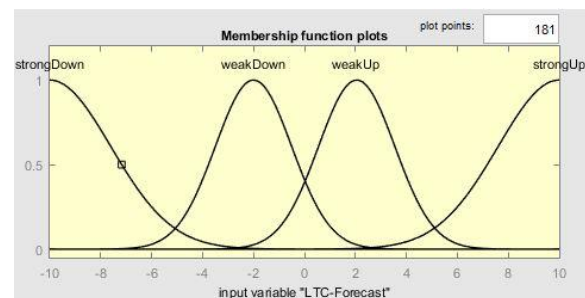


Figure 4: Forecast trend membership functions.

The intersection between the functions depicting weak changes is larger than the intersection between weak and strong functions due to changes around 0% being small and therefore their impact on the price being close to negligible. Fourth input is the trade volume of the Litecoin cryptocurrency.

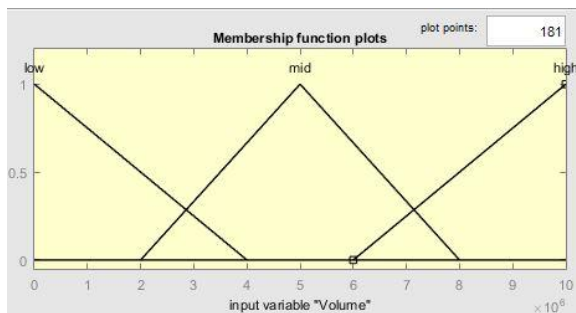


Figure 5: Volume membership functions.

The volume input is modelled with 3 triangular membership functions, *low*, *medium* and *high*, and is defined on range from 0 to 10^7 as shown in Figure 6.

The output is also defined with 3 triangular membership functions, with each function representing one potential action that can be performed, which is to *buy* the cryptocurrency, to *sell* it, or to *hold*, which is to not perform any action.

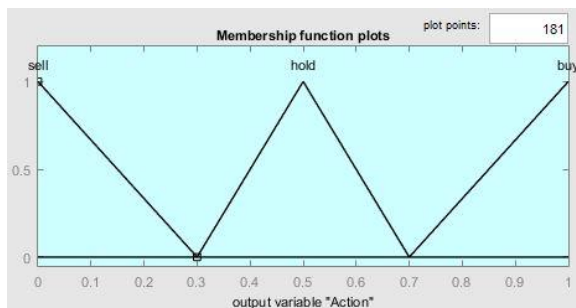


Figure 6: Action membership functions.

As described in Section 2.1, fuzzy rules are modelled in the IF-THEN manner. Given that the FIS structure has 4 inputs, 3 of them having 4 membership functions and 1 having 3 membership functions, this results in 192 possible rules which include all possible input combinations. We will not present all those rules here, but will instead give a brief summary of how they are created. Each rule has a weight factor (WF), which has a value between 0 and 1, with 1 being default value, and indicating maximum weight applied for the rule. Each input value impacts WF in a certain way. If the volume factor is high, WF remains unchanged. If volume is medium, WF is reduced by 0.2, and if volume is low, WF is reduced by 0.4. The logic behind this is that, the higher the volume, there is more certainty that the forecasted direction of the cryptocurrencies is correct. (e.g. if all 3 cryptocurrencies are predicted to go up, this will more likely to happen with the high trade volume than with the low one, due to impact of single trade being less). If the trend of the main cryptocurrency (LiteCoin) is predicted to be strong (either up or down), WF remains unchanged. If the trend is predicted to be low, WF is reduced by 0.2. Similar logic is applied to the BitCoin and Ethereum inputs, with the only difference being that the impact on WF is 0.1 in case of the low trend. Finally, the direction of the predicted trends impacts the direction of the predicted output in a

following manner: if all 3 cryptocurrencies are predicted to go in the same direction, the output will be appropriate action (buy if the trends are up or sell if the trends are down). The same logic is applied if the 2 out of 3 cryptocurrencies have the same predicted trend and 1 of those 2 cryptocurrencies is the LiteCoin. However, due to the third cryptocurrency trend going in the opposite direction, we will reduce the weight of such rule. Finally, if LiteCoin is predicted to go in the different direction than the other 2 cryptocurrencies, the resulting output action is to hold. Some of the rules are presented in Figure 8:



Figure 7: Fuzzy rules

5 Results

First, we present the results for predicting LiteCoin time series using *F* and Fourier transforms without applying FIS on the predicted results.

For the *F*-transform, we tested the model with the following parameter values: window size from 5 to 21 days, *h* value from 0.4 to 1.2, and the order of the polynomial used for regression from 1 to 5. Best results are obtained for the window size of 7 days, *h* value of 0.6 and the polynomial order of 1, meaning that linear regression was used for calculating forecasted *F*-value. The forecasting success rate of *F*-transform predicting whether to *buy* or *sell* the cryptocurrency is 57%.

For the Fourier transform, results were also the best for the window size of 7 days and with usage of linear regression. The forecasting success rate of Fourier transform predicting whether to buy or sell the cryptocurrency is 54%.

As seen on the Figure 1, price of the LiteCoin almost does not change, and volatility is near zero in the period from 8.7.2015 to 4.7.2017. This is the period when correctly predicting the price movement is difficult, and therefore fuzzy rules are applied for such periods with low volatility resulting in hold action when it is not clear whether the trend is up or down.

The second reason for applying FIS and fuzzy rules is that cryptocurrencies have a high correlation, as said in the Section 3, and therefore, prediction of LiteCoin time series can be improved using the historical BitCoin and Ethereum time series. These two cryptocurrencies are more traded, and therefore if they have strong up or down trends, there is a higher probability that LiteCoin will follow that trend.

After applying FIS model described in Section 4.2 on forecasted relative changes in price, the forecasting

success rate increased from 57% to 66% when inputs for FIS are obtained using *F*-transform and from 54% to 61% when inputs for FIS are obtained using Fourier transform. For the low volatility period, the percentage of *hold* action was close to 64% in both cases. All the results described above are summarized in the Table 1.

	<i>F</i> -Transform	Fourier Transform
Without FIS	57	54
With FIS	66	61

Table 1: Forecast success rate [%]

Applying FIS improved the forecasting success rate by 16% when inputs obtained using *F*-transform were used and by 13% when inputs obtained using Fourier transform were used. The improvement is mostly due to ignoring the incorrect predictions in the low volatility period. These percentages can be improved in several ways. First, membership functions can be improved by either changing the shape, or by better calibration of the inflection points. Second, some of the rules which do not contribute to the successfulness of the model can be identified and removed, as having 192 rules in a relatively simple FIS model is too much and the model can be improved by reducing the number of rules. Third improvement which can be done is defining a trend in the time series in stricter manner. Finally, additional inputs can be added to the FIS, such as impact of recent news related to the cryptocurrencies.

On the other hand, the danger of improving the model too much is in over-training the model for the particular time series. We wanted to keep the model flexible and intuitive, and having 66% success rate with a simple *F*-transform + FIS model is reasonably good. In papers to come, we will investigate whether the model can be improved by using neural networks while at the same time keeping its simplicity and flexibility.

6 Conclusion

In this paper we presented a model which combines the time series forecasting ability of the *F* and Fourier transforms with the FIS ability to simulate the expert decision-making system using fuzzy logic rules. The rules were created with purpose to lower the impact of the low volatility periods when it is especially hard to detect the trend, and to exploit the high correlations between the cryptocurrency time series.

Firstly, the forecasting of Litecoin time series is performed solely by means of Fourier or *F* transform. The window size used for forecasting was 6 days, and the price was predicted for the 7th day. Linear regression was used for both computing the forecasted *F*-value used for IDFT and for the polynomial fitting

for the Fourier transform as using polynomials of higher order did not yield better results. The *F*-transform success rate in forecasting 1230 Litecoin prices across 3.5-year period is 57%, while the Fourier transform success rate is a bit lower, 54%. We therefore conclude that in case of analyzed cryptocurrency, the *F*-transform is slightly better than the Fourier transform, which is to be expected, as Fourier transform is mostly used for predicting derivatives prices.

To enhance prediction accuracy, a hybrid approach that combines Fourier or *F* transform with a fuzzy inference system was further considered in the paper. The FIS system used the forecasted relative changes in time series as inputs and outputs appropriate trade action, buy, sell, or hold. The three out of four inputs are forecasted relative changes in price for the 3 cryptocurrencies, while the last one is the trade volume for the Litecoin, which is the cryptocurrency whose price is forecasted. There are 192 rules which are used in FIS. Using FIS increased forecasting success rate by 16% when used with *F*-transform and by 13% when used with Fourier transform.

Some of the ways to improve the model are to check whether adding more inputs will yield better results. Some possible inputs which we are going to test in the following research are the impact of the news, both positive and negative, related to cryptocurrencies. It is also possible that some of the rules defined in FIS are redundant or even have negative effect. Therefore, determining which of the rules should be removed can be a straightforward attempt to improve the model forecasting success rate.

In future research, we will keep the *F*-transform as a forecasting tool, but we will include the neural network as a tool for predicting the *F*-values which will be later on used for IDFT.

References

- [1] S.L. Ho, M. Xie, T.N. Goh, A Comparative Study of Neural Network and Box-Jenkins ARIMA Modeling in Time Series Prediction, *Computers & Industrial Engineering* 42, 2002, pp. 371-375
- [2] P. Zhang, Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model, *Neurocomputing* 50, 2003, pp. 159-175
- [3] J. Contreras, R. Espinola, F.J. Nogales, A. J. Conejo, ARIMA models to predict next-day electricity prices, *IEEE Transactions on Power Systems* 18, 2003, pp 1014-1020
- [4] V. Novák, M. Štěpnička, I. Perfilieva, V. Pavliska, Analysis of periodical time series using soft computing methods, *Computational*

- Intelligence in Decision and Control, 2008, pp. 55-60
- [5] R. J. Frank, N. Davey, S. P. Hunt, Time Series Prediction and Neural Networks, *Journal of Intelligent and Robotic Systems* 31, 2001, pp-91-103
- [6] Y. Gao, M. J. Er, NARMAX Time Series Model Prediction: Feedforward and Recurrent Fuzzy Neural Network Approaches, *Fuzzy Sets and Systems* 150, 2005, pp. 331-350
- [7] M. Štěpnička, O. Polakovič, A Neural Network Approach to the Fuzzy Transform, *Fuzzy Sets and Systems*, 2009, pp. 1037-1047
- [8] I. Luna, I. G. Hidalgo, P. S. M. Pedro, P. S. F. Barbosa, A. L. Francato, P. B. Correia, Fuzzy Inference Systems for Multi-Step Ahead Daily Inflow Forecasting, *Pesquisa Operacional* 37, 2017, pp. 129-144
- [9] Tarno, Subanar, D. Rosadi, Suhartono, Analysis of Financial Time Series Data Using Adaptive Neuro Fuzzy Inference System (ANFIS), *International Journal of Computer Science Issues* 10, 2013
- [10] P. Masset, Analysis of Financial Time-Series using Fourier and Wavelet Methods, *SSRN Electronic Journal*, 2008.
- [11] B. Stádník, J. Raudeliūnienė, V. Davidavičienė, Fourier analysis for stock price forecasting: assumption and evidence, *Journal of Business Economics and Management*, 2016, pp. 365-380
- [12] I. Perfilieva, P. Hod'áková, Fuzzy and Fourier Transforms, *EUSFLAT-LFA*, 2011.
- [13] V. Novák, Reasoning about Mathematical Fuzzy Logic and its Future, *Fuzzy Sets and System*, 2009, pp. 25-44
- [14] M. Štěpnička, Fuzzy Transform for Practical Problems, *Intelligentní systémy pro praxi*, 2007
- [15] M. Štěpnička, V. Pavliska, V. Novák, I. Perfilieva, L. Vavříčková, I. Tomanová, Time Series Analysis and Prediction Based on Fuzzy Rules and the Fuzzy Transform, *IFSA-EUSFLAT 2009 Conference*.
- [16] M. Štěpnička, A. Dvořák, V. Pavliska, L. Vavříčková, Linguistic Approach to Time Series Analysis and Forecasts, *IEEE World Congress on Computational Intelligence*, 2010.
- [17] I. Perfilieva, V. Pavliska, M. Vajgl, B. De Baets, Advanced Image Compression on the Basis of Fuzzy Transforms, *IPMU*, 2008.
- [18] V. Novák, I. Perfilieva, Forecasting Direction of Trend of a Group of Analogous Time Series Using F-Transform and Fuzzy Natural Logic, *International Journal of Computational Intelligence Systems* 8, 2015, pp. 15-28
- [19] F. Di Martino, V. Loia, I. Perfilieva, S. Sessa, An image coding/decoding method based on direct and inverse fuzzy transforms, *International Journal of Approximate Reasoning*, 2008, pp. 110-131
- [20] I. Perfilieva, M. Holcapeka, V. Kreinovich, A new reconstruction from the F-transform components, *Fuzzy Sets and Systems* 288, 2016, pp. 3-25
- [21] M. Štěpnička, R. Valášek Numerical Solution of Partial Differential Equations with Help of Fuzzy Transform, *FUZZ-IEEE*, 2005.
- [22] V. Novák, A. Dvořák, Fuzzy Logic: A Powerful Tool for Modelling of Vagueness, *International Journal of General Systems*, 2008
- [23] V. Novák, S. Lehmke, Logical Structure of Fuzzy IF-THEN Rules, *Fuzzy Sets and Systems*, 2005, pp. 2003-2029
- [24] V. Novák, I. Perfilieva, On the Semantics of Perception-Based Fuzzy Logic Deduction, *Fuzzy Sets and Systems*, 2004, pp. 1007-1031
- [25] R. Bělohávek, V. Novák, Learning Rule Base in Linguistic Expert Systems, *Soft Computing*, 2002, pp. 79-88
- [26] I. Perfilieva, Fuzzy Transforms: Theory and applications, *Fuzzy Sets and Systems* 157, 2006, pp. 993-1023
- [27] A. Dvořák, V. Novák, I. Perfilieva, *Insight into fuzzy modeling*, Wiley, 2016
- [28] S. Pudasaini, Computation of IDFT through forward DFT, 2017
- [29] I. Perfilieva, V. Novák, V. Pavliska, A. Dvořák, M. Štěpnička, Forecasting Time Series Using Fuzzy Transform Forecasting Time Series Using Fuzzy Transform, 2008