Time Series Modelling with Fuzzy Cognitive Maps
Study on an Alternative Concept's Representation Method

Wladyslaw Homenda[1,2], Agnieszka Jastrzebska[1] and Witold Pedrycz[3,4]
[1] *Faculty of Mathematics and Information Science, Warsaw University of Technology, ul. Koszykowa 75, 00-662 Warsaw, Poland*
[2] *Faculty of Economics and Informatics in Vilnius, University of Bialystok, Kalvariju G. 135, LT-08221 Vilnius, Lithuania*
[3] *System Research Institute, Polish Academy of Sciences, ul. Newelska 6, 01-447 Warsaw, Poland*
[4] *Department of Electrical & Computer Engineering, University of Alberta, Edmonton, T6R 2G7 AB, Canada*

Keywords: Fuzzy Cognitive Map, Time Series Modelling and Prediction, Fuzzy Cognitive Map Design.

Abstract: In the article we have discussed an approach to time series modelling based on Fuzzy Cognitive Maps (FCMs). We have introduced FCM design method that is based on replicated ordered time series data points. We named this representation method history_h, where h is number of consecutive data points we gather. Custom procedure for concepts/nodes extraction follows the same convention. The objective of the study reported in this paper was to investigate how increasing h influences modelling accuracy. We have shown on a selection of 12 time series that the higher the h, the smaller the error. Increasing h improves model's quality without increasing FCM's size. The method is stable - gains are comparable for FCMs of different sizes.

1 INTRODUCTION

Fuzzy Cognitive Maps are fairly popular knowledge representation scheme based on weighted directed graphs. Nodes in the graph represent phenomena, arcs represent relations. Among many fields of application, FCMs are used to model time series.

Time series processing with FCMs has a substantial advantage over classical approaches. FCMs operate on the level of concepts, which are information granules. Classical methods are based on scalars. Concepts are higher-level interface for the underlying information. They are introduced to facilitate superior human-machine interactions. Concepts in FCMs are typically realized as a pair of a linguistic description and a fuzzy set. To cover the same fragment of knowledge we either propose a few general concepts or many specific concepts. The level of granulation depends on the modelling objective. General concepts form smaller models, knowledge is highly aggregated. Such models are easier to interpret and to use, but at the same time they are less precise in a numerical context. In contrast, many specific concepts produce larger model. Hence, predictions are more accurate, but we may loose ease of interpretation.

We would like to emphasize that the goal of time series modelling with FCMs is to offer human-centered models. We take advantage of concepts-based information representation scheme to describe complex phenomena in as intuitive as possible way.

Study presented in this paper is focused on FCM design procedure for time series modelling. In our previous works we have discussed a general approach to FCM design, and here we investigate its crucial element: time series representation. The objective of our research was to experimentally test how increasing time span represented by concepts would influence modelling quality. The research discussed in this paper aims at improvement of FCM design so that one can achieve higher precision and maintain model's intuitiveness and ease of interpretation at the same time.

Proposed FCM-based time series modelling is novel and not yet present in the literature.

The article is structured as follows. Section 2 summarizes relevant research to be found in the literature. Section 3 presents developed methods. Section 4 contains a description of experiments' results. Section 5 covers conclusion and future research directions.

2 LITERATURE REVIEW

The research on FCMs has its beginnings in 1986, when B. Kosko published his work in (Kosko, 1986).

Homenda W., Jastrzebska A. and Pedrycz W..
Time Series Modelling with Fuzzy Cognitive Maps - Study on an Alternative Concept's Representation Method.
DOI: 10.5220/0005207704060413
In *Proceedings of the International Conference on Agents and Artificial Intelligence* (ICAART-2015), pages 406-413
ISBN: 978-989-758-074-1

FCMs are present in sciences for 30 years now, however, their application to time series modelling has been in the scope of interest for less than a decade.

There are three distinct approaches to time series modelling with FCMs considered in the literature. First, and most commonly studied one has been introduced by W. Stach, L. Kurgan, and W. Pedrycz in (Stach et al., 2008) in 2008. Cited work introduced FCM design method for time series modelling based on:

- time series representation as a sequence of pairs: amplitude, change of amplitude,
- scalar time series fuzzification and extraction of concepts that represent (fuzzified) time series values; concepts directly correspond to nodes in the FCM,
- real-coded genetic algorithms for FCM training.

Summarized research has been an inspiration for further attempts in this area. Examples of articles presenting a research in the same direction are: (Lu et al., 2013) and (Song et al., 2010).

Second approach to time series modelling with FCMs has been brought up by W. Froelich and E. I. Papageorgiou (Froelich et al., 2012; Froelich and Papageorgiou, 2014). Proposed method is dedicated to multivariate time series. In this approach FCM's nodes correspond to variables of the time series.

Third method has been introduced by the authors in (Homenda et al., 2014b). This approach is based on moving window technique. Nodes in a FCM are ordered and they represent consecutive time points.

In this article we follow the approach introduced in (Stach et al., 2008). Common elements of our and cited method are: fuzzy concepts, fuzzification of observations into membership values. What is different: time series and concepts representation method and application of FCM simplification. We also use different training procedure, but this is only a technical part of the modelling process.

3 DESIGNING FUZZY COGNITIVE MAPS FOR TIME SERIES MODELLING

The objective of the research on the application of FCMs to time series modelling is to propose a design technique that will extract interpretable nodes and provide necessary training data to train the weights' matrix.

Methods developed and investigated for this article are related to the original approach introduced in (Stach et al., 2008). In general, the proposed approach and experiments on time series modelling with FCMs could be decomposed into the following phases:

1. Time series conceptual representation.
2. FCM training.
3. Time series prediction (on a conceptual level).

3.1 Time Series Representation for Modelling with Fuzzy Cognitive Maps

Typically, a time series we process is not in conceptual form, but in scalar uni- or multivariate form. The reason is that we often use instruments, such as thermometers, seismometers, etc. to report current states of phenomena. Conceptual knowledge repositories are not available at all. This is why first we discuss an algorithmic model for transforming any scalar time series to conceptual one.

Proposed method is valid for both multi- and univariate time series. Here for illustration purposes we use univariate ones.

Let us denote a scalar time series as below:

$$a_1, a_2, a_3, \ldots, a_M, \text{ where } a_i \in \mathbb{R} \text{ for } i = 1, 2, \ldots, M$$

The sequence above is the most basic representation scheme for a time series. In our model we use repeated historical data points scheme, which we call history_h, $h \in \mathbb{N}, h \leqslant M$. Depending on the time span h, elements of the original time series get repeated with preservation of their order. For example:

- for history_2 time series is represented as follows:
$$(a_2, a_1), (a_3, a_2), (a_4, a_3), \ldots, (a_M, a_{M-1})$$
- for history_3 time series is represented as follows:
$$(a_3, a_2, a_1), (a_4, a_3, a_2), \ldots, (a_M, a_{M-1}, a_{M-2})$$

And so on. Each element from the time series is repeated h times (the only exception is for first $h-1$ elements). This time series representation model does not pre-process the original time series. It just repeats already available information. In contrast, method proposed by W. Stach et al. is based on time series representation with pairs of amplitude and change of amplitude. Theoretically, Stach's et al. approach is equivalent to history_2, but in our previous research (Homenda et al., 2014a) we have shown that in practise unprocessed time series values are better: they give models of higher accuracy and are easier to interpret.

The greater the h, the more complex time series representation. Let us take a closer look at $h = 3$ (history_3). Such model is very easy to interpret for a human being. Each triple (a_{i-2}, a_{i-1}, a_i) is interpreted as (day before yesterday, yesterday, today) if the interval for gathering information was a day. In general, history_3 is describing triples of (before before, before, now) data points. In comparison, Stach's et. al method in its generalized equivalent would be representing the time series as triples of: (amplitude, change of amplitude, change of amplitude change). The higher the h, the less straightforward Stach's model gets, while the method investigated in this paper still maintains its transparency and is easy to interpret.

Note that a time series represented with the history_h model for $h = 1,2,3$ can be plotted in 1-,2-,3-dimensional coordinates systems respectively. h can be treated as the number of system's dimensions.

Figure 1 illustrates an exemplar real-world time series named Kobe in 1-dimensional space in time (first plot), in 2-dimensional space of present and past (middle image), and in 3-dimensional space of current, past and before past. Kobe time series is publicly available in a repository under (Hyndman, 2014).

3.2 Extraction of Concepts

Let us now discuss a method for elevation of a scalar time series to a higher abstract level of concepts. We aid ourselves, similarly as W. Stach et al., with Fuzzy C-Means algorithm. The objective is to propose c concepts that generalize the time series. In our procedure it is enough to perform it once, because selected time series representation scheme uses repeated, but unprocessed values. For Stach's et al. method it is necessary to perform it h times. The aim of applying Fuzzy C-Means to the original time series $a_0, a_1, \ldots$ is to detect c clusters' centers that become concepts' centres. Concepts describe in a granular, aggregated fashion numerical values of the time series. The value of c is determining model's specificity.

The number of target concepts can be either up to system's designer or decided by appropriate procedures. An example of such procedure is described in (Pedrycz and de Oliveira, 2008). Accuracy of clustering into concepts is a very important factor. On the other hand, the value of c is directly influencing model's complexity. High values of c produce large and harder to interpret models. FCM-based model's quality is first of all judged by it's ease of interpretation and application.

Concepts generalize values of the time series. The choice of concepts is followed by a selection of appropriate linguistic variables that are attached to them. Here are examples of linguistic variables: Small, High, Small Negative, Moderately High, etc. For convenience we usually abbreviate them and use only first letters of each word. Linguistic variables enhance the model and provide intuitive interface for its final users - humans.

In our previous research we have investigated the issues of the balance between specificity and generality of the model. We have shown that to some extent it depends on the character of a time series. In this paper for comparability we assume $c = 3$ in all experiments and attach the following linguistic variables: Small (S), Moderate (M), High (H). Other values of c are not considered here, because the main focus is on time series representation. $c = 3$ was selected, because of its simplicity.

Moreover, Fuzzy C-Means not only extracts concepts' centres, but also provides a formula for calculating membership levels (u) to proposed concepts:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{||\mathbf{a}_i - \mathbf{v}_j||}{||\mathbf{a}_i - \mathbf{v}_k||} \right)^{2/(m-1)}} \tag{1}$$

u_{ij} is a membership value of i-th data point to j-th concept. c is the number of clusters-concepts, so $j = 1, \ldots, c$. Data points in this context match time series representation scheme. Data points are denoted as $\mathbf{a}_i = [a_{i+h-1}, \ldots, a_i], i = 1, \ldots, M - h + 1$, where M is the length of the input scalar time series. $\mathbf{v}_j = [v_{j1}, \ldots, v_{jh}]$ describes j-th concept. m is fuzzification coefficient ($m > 1$), $||\cdot||$ is the Euclidean norm,

On the output of Fuzzy C-Means procedure we obtain 1-dimensional concepts' centres. To adapt them to the time series representation scheme we have to elevate them to h-dimensional space by applying h-nary Cartesian product. This results in c^h new h-dimensional concepts.

Again, with the use of Formula 1 we calculate membership levels of time series data points to new concepts. This time the number of concepts is not c, but c^h.

Concepts and values of corresponding membership levels for time series data points provide all necessary information to train a FCM. Extracted concepts become nodes in FCM. Training data are levels of membership, which during the process of FCM learning are passed to appropriate nodes.

Proposed method for FCM design purposefully extracts a lot of concepts. We apply Cartesian product, so we suspect that not all of the proposed concepts reflect the underlying time series. We may evaluate quality of a concept $\mathbf{v}_j$ by calculating its membership index $M(\mathbf{v}_j)$:

$$M(\mathbf{v}_j) = \sum_{i=1}^{N} u_{ij} \tag{2}$$

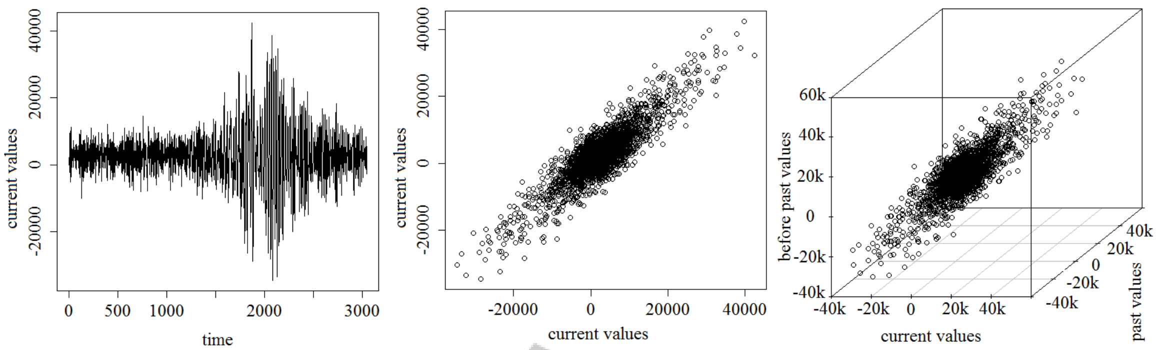

Figure 1: Kobe time series and its representation in: history_1 (left plot), history_2 (middle plot), and history_3 (right plot).

N is the size of training data. Training data is a subset of all available data used in FCM design and training. The remainder data subset is called test and it is for evaluation of the proposed model.

Membership index informs how strongly a concept represents the time series. $M(\mathbf{v}_j)$ is a sum of membership values of all data points to concept j. Weak values of membership index say that given concept is weakly tied to the dataset. Strong values of membership index indicate that the concept is able to represent time series well.

In the process of further model tuning one may decide to get rid of weak nodes. Hence, we introduce letter n to describe the final number of concepts in the model. Membership index can be used to separate good nodes from bad ones. We call this procedure FCM simplification as its objective is to make the model smaller. In this paper we present a series of experiments, where we apply membership index to obtain FCMs of desired size.

3.3 Learning Procedure

FCMs represent relations within knowledge. FCM size - the number of its nodes is denoted as n. FCMs are illustrated with weighted directed graphs. The core of each FCM is its weights' matrix denoted as $\mathbf{W} = [w_{ij}], w_{ij} \in [-1,1], i,j = 1,\ldots,n$. Weights describe relations between modelled phenomena. Though fuzzy sets are associated with the $[0,1]$ domain, weights values span onto the whole $[-1,1]$ domain. In this way we are able to represent negative relations. Weights that are evaluated as 0 mean that there is no relation between two concepts. In the case of the proposed time series modelling scheme, nodes in the FCM are equivalent to concepts. Hence, nodes represent concepts that generalize values of the time series.

FCM training procedure aims at optimizing weights matrix $\mathbf{W}$. The weights matrix learning procedure's objective is to minimize differences between predictions provided by the FCM and real, target values of conceptualized time series.

FCM exploration is by the following formula:

$$\mathbf{Y} = f(\mathbf{W} \cdot \mathbf{X}) \tag{3}$$

where f is a sigmoid transformation function:

$$f(t) = \frac{1}{1 + exp(-\tau t)} \tag{4}$$

Value of parameter τ was set to 5 based on experiments and literature review: (Mohr, 1997; Stach et al., 2008).

In the input layer of the FCM-based model we have activations ($\mathbf{X} = [x_{ji}], x_{ji} \in [0,1], j = 1,\ldots,n, i = 1,\ldots,N$). N is the number of available training data. Single activations vector is denoted as $\mathbf{x}_i = [x_{1i}, x_{2i}, \ldots, x_{ni}]$ and it concerns i-th data point $\mathbf{a}_i$. In the FCM exploration process, activations are passed to FCM nodes.

In the output layer of FCM-based model we have FCM responses denoted as $\mathbf{Y} = [y_{ji}], y_{ji} \in [0,1], j = 1,\ldots,n, i = 1,\ldots,N$. FCM responses model and predict phenomena of interest. Hence, the quality of given FCM is assessed with discrepancies between FCM responses $\mathbf{Y}$ and goals $\mathbf{G}$.

Goals matrix $\mathbf{G} = [g_{ji}], g_{ji} \in [0,1]$ is of size $n \times N$ and it gathers actual, reported states of n nodes, N is the number of training data.

FCM learning procedure iteratively adjusts weights matrix, so that the aforementioned discrepancies are minimized. Typically, objective function is Mean Squared Error (MSE) between FCM outputs and goals:

$$MSE = \frac{1}{n \cdot N} \cdot \sum_{i=1}^{N} \sum_{j=1}^{n} (y_{ji} - g_{ji})^2 \tag{5}$$

In the literature one may find many articles devoted to FCMs learning. Since this is not the subject of this paper we do not discuss this research to greater extent here, just give references to selected papers on different approaches: (Papageorgiou et al., 2003; Parsopoulos et al., 2003; Stach et al., 2005). In our experiments we train FCMs with the use of PSO, with its implementation in R available in "pso" package. All arguments were left to default Standard PSO 2007, full list is under (Bendtsen, 2014).

4 EXPERIMENTS

4.1 Study Objectives and Methods

The focus of this paper is on the proposed time series representation scheme, named history_h. The objective of our study is to investigate how increasing the length of time span (h) represented by concepts improves modelling accuracy. We are especially interested if there is any differences in gains for small and large FCMs.

In order to address named issues we have set up a series of experiments. We have investigated 3 synthetic and 9 real-world time series. Synthetic time series were constructed based on a fixed base of numbers that constitute their period. Base sequence was replicated so that the total length of the input time series is 3000. Next, we added a noise: to each time series scalar data a random value from a normal distribution having a mean of 0 and a standard deviation of 0.7. In further parts of this article we name synthesised time series by their base sequence.

Real-world time series were downloaded from a publicly available repository under (Hyndman, 2014). Names of real-world time series in this article are the same as in the cited database. Selected time series were of significantly different properties. Due to space limitations we do not elaborate on their characteristics. The aim of using both synthetic and real-world time series was to thoroughly test the proposed time series representation scheme. Synthetic time series are very regular, while real-world datasets are not.

The results presented in this paper are a selection of an extensive series of experiments on more time series. Due to space limitations we use just 12 datasets to illustrate our approach, results of other tests were consistent with the ones discussed here.

In the course of the experiments we have used first 70% of data points for FCM learning, remaining 30% were for predictions only. First part is called train, second test set. Experiments were for FCMs of different sizes: $n = 4, 6, 8, 10, 12, 17, 22, 27$. For each case we have tested history_3, history_4, and history_6 representation schemes.

Note that in order to compare results of the three time series representation methods maps had to be of the same size. Therefore, we have applied membership index to remove appropriate number of worst nodes from models that were originally large. Models obtained with history_3 method were originally of size $3^3 = 27$. Models with history_4 were of size $3^4 = 81$. Models with history_6 had initially $3^6 = 729$ concepts. The largest of trained maps had $n = 27$ nodes, which is a lot, but still manageable to train and apply. Large models are impractical to train and to apply. Let us remind that the number of weights for optimization is the number of concepts squared, but time necessary to train such FCM grows much faster than quadratically. Therefore, we have determined that $n = 27$ is one of largest reasonable FCM. Its weights matrix requires around 3-4 days to optimize with a single-threaded process on a standard PC.

As a measure of quality we use MSE, which has been also the objective function for optimization procedure. Figure 2 illustrates MSE for 3 synthetic and 9 real-world time series. Chart is ordered according to FCM size. Names of time series are in plots, the following abbreviations are used: h3 for history_3, h4 for history_4, and h6 for history_6.

4.2 Results

The accuracy of modelling depends on map size. The larger the map, the smaller the MSE. This observation comes with no surprise though. In large maps concepts generalize smaller fragment of information. In contrast, small maps, like $n = 4$ are based on general concepts that have to represent greater part of the time series, hence model's accuracy is worse.

Proposed time series representation scheme allows to increase the accuracy of models without increasing FCM size. In each case the greater h, the smaller the MSE for FCMs of the same size. MSEs on conceptualized real-world time series are typically lower than on synthetic time series. Observe, that the OY scale for the best fitted models is to 0.004. Errors for four such time series: Synthetic-258, DailyIBM (stocks), Equiptemp (temperature of lab equipment), Wave2 (frequency of waves) are very small. For the DailyIBM time series errors were so small that the bars are hardly visible in used scale.

In Figure 2 we see that for smaller maps, like $n = 4$ and $n = 6$, increasing h resulted in greater improvement than for large maps. This is natural consequence of the fact that errors on smaller maps were higher at start, so we can improve them more.

It is worth to notice, that errors on predictions (test) are very close to errors on training data. Test

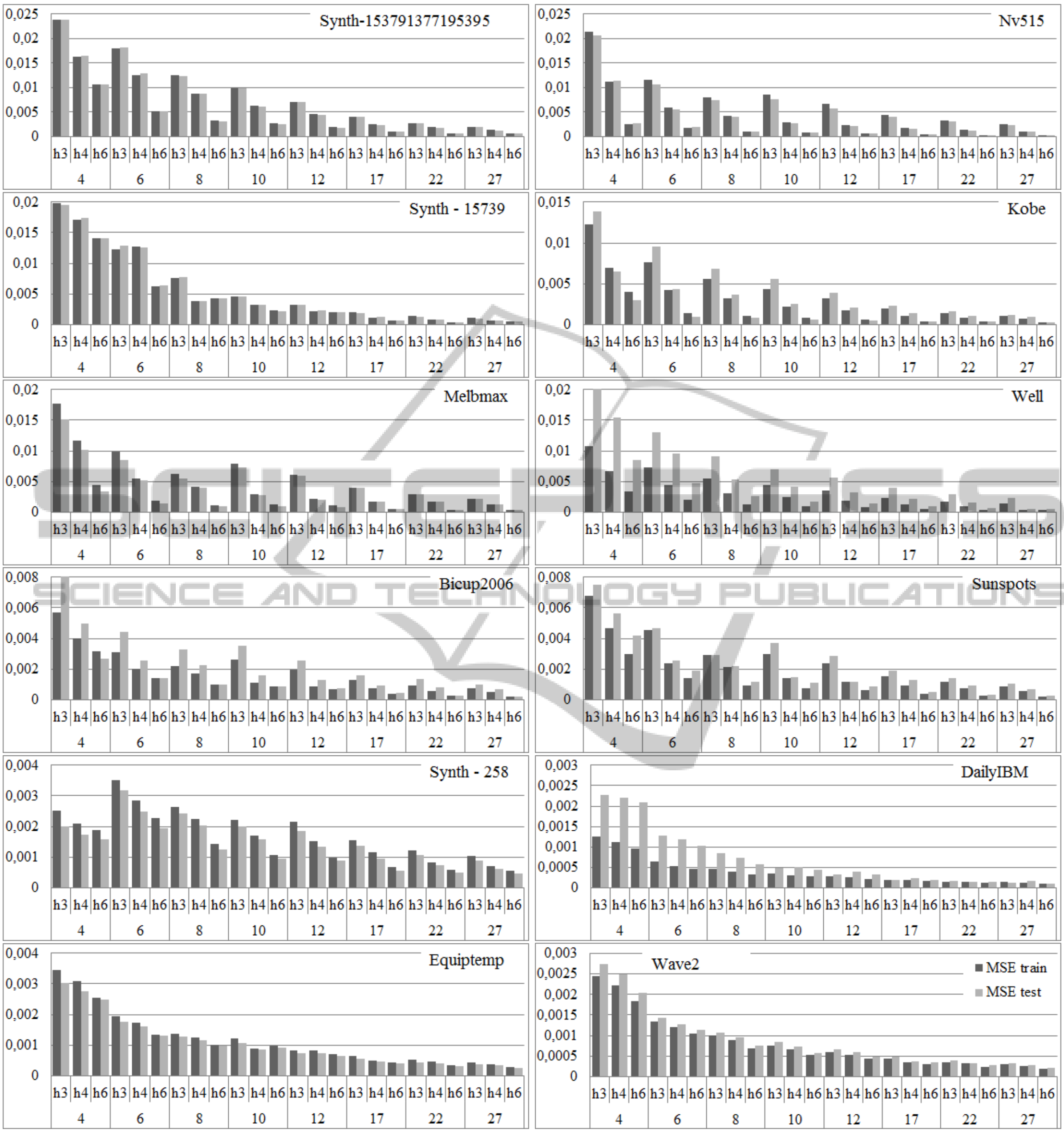

Figure 2: MSE for 3 synthetic and 9 real-world fuzzified time series for FCM-based models with different number of nodes (4, 6, ..., 27) and different time series representation schemes: h3 stands for history_3, h4 for history_4 and h6 for history_6.

predictions were for data points that were not included in the training dataset.

Let us now investigate the gain of accuracy achieved by applying time series representation scheme with larger h. Figure 3 illustrates the gain in MSE achieved by increasing value of h from 3 to 4 (lighter bars) and from 3 to 6 (darker bars).

Plots in Figure 3 show percentage increase of modelling accuracy. Gains were calculated based on a percentage decrease of MSE, achieved when we increase h. We have used the following formula:

$$Gain_{h+d} = \frac{MSE_h - MSE_{h+d}}{MSE_h} \cdot 100\% \quad (6)$$

$Gain_{h+d}$ informs about a percentage decrease of MSE when we increase the time span from h to $h+d$. Negative values of Gain inform that there was no increase at all. The larger the Gain, the more MSE dropped by

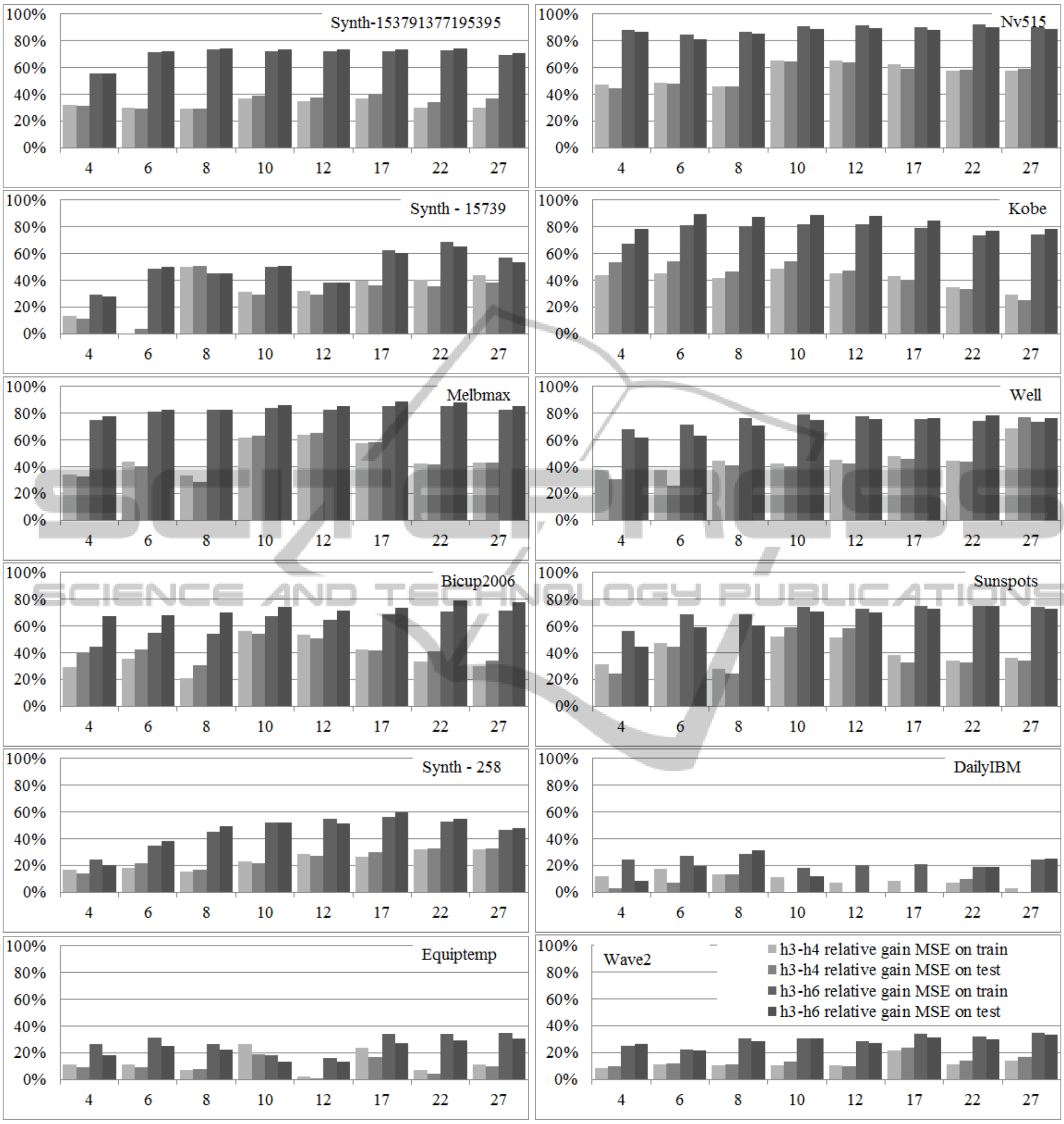

Figure 3: Percentage gain of accuracy on train and test sets achieved by increasing the time span: from $h = 3$ to $h = 4$ pale grey bars, and from $h = 3$ to $h = 6$ dark grey bars.

increasing time span.

The more we increase the time span coded in concepts, the higher accuracy we achieve. In Figure 3 lighter bars that correspond to the upgrade from history_3 to history_4 are smaller than darker ones plotted for the upgrade from history_3 to history_6.

The gain is comparable for FCMs of all sizes. Small FCMs can be equally well improved as large ones. This is especially appealing property. By increasing h we are able to achieve better results for very small models without increasing their size.

The gain is different for different time series. For three time series that were modelled with very high precision at start (for history_3): DailyIBM, Equiptemp, and Wave2 the improvement is lower than for the other time series.

To sum up, the proposed time series representation method improves modelling quality. Typically, increasing the time span for concepts from $h = 3$ to $h = 4$ improves modelling accuracy by circa 40%, while increasing the time span from $h = 3$ to $h = 6$ improves it by circa 80%.

5 CONCLUSIONS

Proposed time series and concepts' representation method for time series modelling with FCMs is based on gathering h consecutive elements of the input time series into a single data point. The larger the h, the longer time span is captured in a single data point and in extracted concepts. Empirical studies show that applying this method is beneficial. It allows to increase modelling accuracy without increasing FCM size.

We have shown that long time spans (like $h = 6$) bring higher numerical accuracy, but in our opinion for a model that would be used by humans $h = 3$ or 4 are reasonable values. The larger h, the more computationally demanding is the modelling process. The allocation of a matrix for membership values is memory-demanding, while optimization of weights matrix is time-demanding.

Most important advantage of the proposed method is transparency of time series and concepts representation method. Each data point in our model has the same interpretation and it represents an h-long time span. Information is not processed and, if necessary, can be translated in a straightforward way back to the numeric time series.

In future research we will address interpretation issues of trained FCMs. We will also take under further investigation FCM training procedure.

ACKNOWLEDGEMENTS

The research is partially supported by the Foundation for Polish Science under International PhD Projects in Intelligent Computing. Project financed from The European Union within the Innovative Economy Operational Programme (2007- 2013) and European Regional Development Fund.

The research is partially supported by the National Science Center, grant No 2011/01/B/ST6/06478.

REFERENCES

Bendtsen, C. (2014). Package 'pso'. http://cran.r-project.org/web/packages/pso/pso.pdf. [Online; accessed 14-November-2014].

Froelich, W. and Papageorgiou, E. (2014). Extended evolutionary learning of fuzzy cognitive maps for the prediction of multivariate time-series. In *Fuzzy Cognitive Maps for Applied Sciences and Engineering*, pages 121–131.

Froelich, W., Papageorgiou, E., Samarinasc, M., and Skriapasc, K. (2012). Application of evolutionary fuzzy cognitive maps to the long-term prediction of prostate cancer. In *Applied Soft Computing 12*, pages 3810–3817.

Homenda, W., Jastrzebska, A., and Pedrycz, W. (2014a). Fuzzy cognitive map reconstruction - dynamics vs. history. In *Proceedings of ICNAAM*, page in press.

Homenda, W., Jastrzebska, A., and Pedrycz, W. (2014b). Modeling time series with fuzzy cognitive maps. In *Proceedings of the WCCI 2014 IEEE World Congress on Computational Intelligence*, pages 2055–2062.

Hyndman, R. (2014). Time series repository. http://robjhyndman.com/tsdldata. [Online; accessed 14-November-2014].

Kosko, B. (1986). Fuzzy cognitive maps. In *Int. J. Man Machine Studies 7*.

Lu, W., Yang, J., and Liu, X. (2013). The linguistic forecasting of time series based on fuzzy cognitive maps. In *Proc. of IFSA/NAFIPS*, pages 2055–2062.

Mohr, S. (1997). The use and interpretation of fuzzy cognitive maps. In *Master's Project, Rensselaer Polytechnic Institute*.

Papageorgiou, E. I., Stylios, C. D., and Groumpos, P. P. (2003). Fuzzy cognitive map learning based on nonlinear hebbian rule. In *Australian Conf. on Artificial Intelligence*, pages 256–268.

Parsopoulos, K., Papageorgiou, E., Groumpos, P., and Vrahatis, M. (2003). A first study of fuzzy cognitive maps learning using particle swarm optimization. In *Proc. IEEE 2003 Congr. on Evolutionary Computation*, pages 1440–1447.

Pedrycz, W. and de Oliveira, J. V. (2008). A development of fuzzy encoding and decoding through fuzzy clustering. In *Clustering, in IEEE Transactions on Instrumentation and Measurement, Vol. 57, No. 4*, pages 829–837.

Song, H., Miao, C., Roel, W., and Shen, Z. (2010). Implementation of fuzzy cognitive maps based on fuzzy neural network and application in prediction of time series. In *IEEE Transactions on Fuzzy Systems, 18(2)*, pages 233–250.

Stach, W., Kurgan, L., and Pedrycz, W. (2008). Numerical and linguistic prediction of time series. In *IEEE Transactions on Fuzzy Systems, 16(1)*, pages 61–72.

Stach, W., Kurgan, L., Pedrycz, W., and Reformat, M. (2005). Genetic learning of fuzzy cognitive maps. In *Fuzzy Sets and Systems 153*, pages 371–401.