Marian B. GORZAŁCZANY, Filip RUDZIŃSKI
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING, KIELCE UNIVERSITY OF TECHNOLOGY
Al. 1000-lecia Państwa Polskiego 7, 25-314 Kielce

Clustering and filtering of measurement data based on dynamic self-organizing neural networks

Prof. Marian B. GORZAŁCZANY

In 1978 he received the M.Sc. degree in electronics from Warsaw Univ. of Technology, in 1983 and 1994 – the Ph.D. and D.Sc. degrees in computer science from Poznań Univ. of Technology and in 2003 – the professor title. In 1990-1992 he was working at Univ. of Saskatchewan and Univ. of Guelph, Canada. He is the author/co-author of more than 80 refereed scientific works including the monograph published by Springer-Verlag. The main area of scientific interests is designing and application of computational intelligence systems.

e-mail: m.b.gorzalczany@tu.kielce.pl

Ph.D. eng. Filip RUDZIŃSKI

He graduated from Department of Electrical and Computer Engineering, Kielce University of Technology (specialization: electrical engineering) in 2000. He received his Ph.D. degree in computer science from Systems Research Institute of Polish Academy of Sciences in 2006. The area of his scientific interests covers artificial intelligence, including neural networks, fuzzy systems and genetic algorithms.

e-mail: f.rudzinski@tu.kielce.pl

Abstract

The paper presents an application of dynamic self-organizing neural networks (introduced by the same authors) to clustering of complex, multidimensional measurement-type data using as an example the so-called *Synthetic Control Chart Time Series* available at WWW server of the Department of Information and Computer Science, the University of California at Irvine. Moreover, after deactivation of some of the mechanisms governing the operation of the proposed networks they become efficient tools for signal and data filtering. The filtering of *Equiptemp* measurement data set available from *Time Series Library* by means of the proposed networks is also briefly presented.

Keywords: computational intelligence, self-organizing neural networks, clustering, filtering, measurement data.

Grupowanie i filtracja danych pomiarowych z wykorzystaniem dynamicznych, samoorganizujących się sieci neuronowych

Streszczenie

Artykuł prezentuje zastosowanie tzw. dynamicznych samoorganizujących się sieci neuronowych (zaproponowanych przez autorów tej pracy) do grupowania złożonych, wielowymiarowych danych pomiarowych na przykładzie zbioru danych *Synthetic Control Chart Time Series* dostępnego na serwerze WWW Uniwersytetu Kalifornijskiego w Irvine (Department of Information and Computer Science). Proponowane sieci, w trakcie procesu uczenia, są w stanie dzielić swoje łańcuchy neuronów na podłańcuchy, ponownie łączyć wybrane podłańcuchy ze sobą oraz dynamicznie zmieniać całkowitą liczbę neuronów sieci. Cechy te umożliwiają im jak najlepsze dopasowanie się do nieznanych z góry struktur "zakodowanych" w danych. Funkcjonowanie proponowanych sieci zilustrowano najpierw na przykładzie złożonego zbioru danych dwuwymiarowych typu dwóch spiral. Po wyłączeniu pewnych mechanizmów rządzących funkcjonowaniem proponowanych sieci stają się one również efektywnymi narzędziami filtracji sygnałów. Przykłady filtracji danych pomiarowych zawartych w zbiorze *Equiptemp* pochodzącym z tzw. *Time Series Library* są również przedstawione w artykule.

Słowa kluczowe: inteligencja obliczeniowa, samoorganizujące się sieci neuronowe, grupowanie, filtracja, dane pomiarowe.

1. Introduction

Data clustering or cluster analysis consists in partitioning a given set of data (observations) into clusters (classes, groups, subsets) so that the elements of each cluster are as "similar" as possible to each other and as "dissimilar" as possible from those of the other clusters. Clustering is usually performed without having any supervisory information regarding the data and, in particular, without knowing in advance the number of clusters in the data set. Efficient clustering techniques provide powerful tools for a high-level intelligent analysis of complex, multidimensional data (including measurement data) and the knowledge discovery in data sets by revealing – in an automatic way – previously hidden, "natural" structures (major trends, patterns, decision mechanisms, etc.) in them in order to support the user in better understanding of the data and thus in making more sensible decisions.

This paper presents an application of dynamic self-organizing neural networks (introduced by the same authors in [3, 4, 5] and some earlier papers such as [1, 2]) to clustering of complex, multidimensional measurement-type data. First, the dynamic self-organizing neural networks are briefly presented. In the course of learning, they are able to disconnect their neuron chains, to reconnect some of the sub-chains again, and to dynamically adjust the overall number of neurons in the system. All these features enable them to fit in the best way the structures "encoded" in data sets in order to display them to the user. The operation of the proposed clustering technique has been illustrated by means of exemplary two-dimensional complex data set. Then, this technique has been applied to clustering of selected complex and multidimensional measurement-type data set (the so-called *Synthetic Control Chart Time Series*) available at WWW server of the Department of Information and Computer Science, the University of California at Irvine [7]. Moreover, after deactivation of some of the mechanisms governing the operation of the proposed networks they become efficient tools for signal and data filtering. The filtering of *Equiptemp* measurement data set available from *Time Series Library* [6] by means of the proposed networks is also briefly presented.

2. Dynamic self-organizing neural networks for data clustering [3, 4, 5]

A dynamic self-organizing neural network is a generalization of the conventional self-organizing neural network with one-dimensional neighbourhood. Consider the latter case of the network that has *n* inputs x_1, x_2,..., x_n and consists of *m* neurons arranged in a chain; their outputs are y_1, y_2,..., y_m, where $y_j = \sum_{i=1}^{n} w_{ji} x_i$, $j = 1,2,...,m$ and w_{ji} are weights connecting the output of *j*-th neuron with *i*-th input of the network. Using vector notation ($\boldsymbol{x} = (x_1, x_2, ..., x_n)^T$, $\boldsymbol{w}_j = (w_{j1}, w_{j2}, ..., w_{jn})^T$), $y_j = \boldsymbol{w}_j^T \boldsymbol{x}$. The learning data consists of *L* input vectors $\boldsymbol{x}_l$ $(l = 1,2,...,L)$. The first stage of any Winner-Takes-Most (WTM) learning algorithm that can be applied to the considered network, consists in determining the neuron $j_{\boldsymbol{x}}$ winning in the competition of neurons when learning vector $\boldsymbol{x}_l$ is presented to the network. Assuming the normalization of learning vectors, the winning neuron $j_{\boldsymbol{x}}$ is selected such that

$$d(\boldsymbol{x}_l, \boldsymbol{w}_{j_x}) = \min_{j=1,2,\ldots,m} d(\boldsymbol{x}_l, \boldsymbol{w}_j), \quad (1)$$

where $d(\boldsymbol{x}_l, \boldsymbol{w}_j)$ is a distance measure between $\boldsymbol{x}_l$ and $\boldsymbol{w}_j$; throughout this paper, the Euclidean distance measure will be applied:

$$d(\boldsymbol{x}_l, \boldsymbol{w}_j) = \| \boldsymbol{x}_l - \boldsymbol{w}_j \| = \sqrt{\sum_{i=1}^{n} (x_{li} - w_{ji})^2}\ . \quad (2)$$

The WTM learning rule can be formulated as follows:

$$\boldsymbol{w}_j(k+1) = \boldsymbol{w}_j(k) + \eta_j(k) N(j, j_{\boldsymbol{x}}, k)[\boldsymbol{x}(k) - \boldsymbol{w}_j(k)], \quad (3)$$

where k is the iteration number, $\eta_j(k)$ is the learning coefficient, and $N(j, j_{\boldsymbol{x}}, k)$ is the neighbourhood function. In this paper, the Gaussian-type neighbourhood function will be used:

$$N(j, j_{\boldsymbol{x}}, k) = e^{-\frac{(j-j_{\boldsymbol{x}})^2}{2\lambda^2(k)}}, \quad (4)$$

where $\lambda(k)$ is the “radius” of neighbourhood (the width of the Gaussian “bell”).

The generalization of the above-presented conventional self-organizing neural network with one-dimensional neighbourhood to the dynamic network consists in introducing mechanisms that allow the network:

a) to automatically adjust the number of neuron in its neuron chain (by removing low-active neurons from the chain and adding new neurons in the areas of high-active neurons in the chain),
b) to automatically disconnect its neuron chain, as well as to reconnect some of the sub-chains again if required.

These two features enable the dynamic self-organizing neural network to fit in the best way the structures that are “encoded” in data sets in order to display them to the user. These features are implemented by activating (under some conditions) – after each learning epoch – five successive operations:

1) the removal of single, low-active neurons,
2) the disconnection of the neuron chain,
3) the removal of short neuron sub-chains,
4) the insertion of additional neurons into the neighbourhood of high-active neurons, and
5) the reconnection of two selected sub-chains of neurons.

The operations nos. 1, 3, and 4 are the components of the mechanism for automatic adjustment of the number of neurons in the chain, whereas the operations nos. 2 and 5 govern the disconnection and reconnection mechanisms, respectively. Based on experimental investigations, conditions have been formulated under which particular operations are activated. Unfortunately, due to a limited space for this publication, we are not able to present them here; the details can be found in [3] and [5].

After the completion of the learning process, the neural chain (composed of sub-chains) determines a kind of “route” in the data space revealing the distribution of clusters in the data space. An analysis of this route (by determining the so-called histogram of nearness between neighbouring neurons along the neuron chain) provides the user with an image – on the plane – of the cluster distribution in the considered multidimensional data set. First, the histogram of distances H_j^{dist} between two neighbouring neurons nos. j and $j+1$ along the neuron chain ($j = 1,2,\ldots,m-1$, m – number of neurons in the chain) is determined as follows

$$H_j^{dist} = d(\boldsymbol{w}_j, \boldsymbol{w}_{j+1}) = d_{j,j+1}, \quad (5)$$

where d is the Euclidean distance measure as in (2). Then, the corresponding histogram of nearness H_j^{near} is calculated

$$H_j^{near} = (\max_{i=1,2,\ldots,m-1} H_i^{dist}) - H_j^{dist} = (\max_{i=1,2,\ldots,m-1} d_{i,i+1}) - d_{j,j+1}\ . \quad (6)$$

The higher the bars of the nearness histogram are, the closer the corresponding neurons are situated and thus, the data they represent belong to more compact clusters.

Fig. 1 shows the performance of the dynamic self-organizing neural network applied to the synthetic data set presented in Fig.1a. The data set contains clusters in the form of two spirals “wound” on each other (the two-spiral data are often used as benchmark data for testing different clustering techniques).

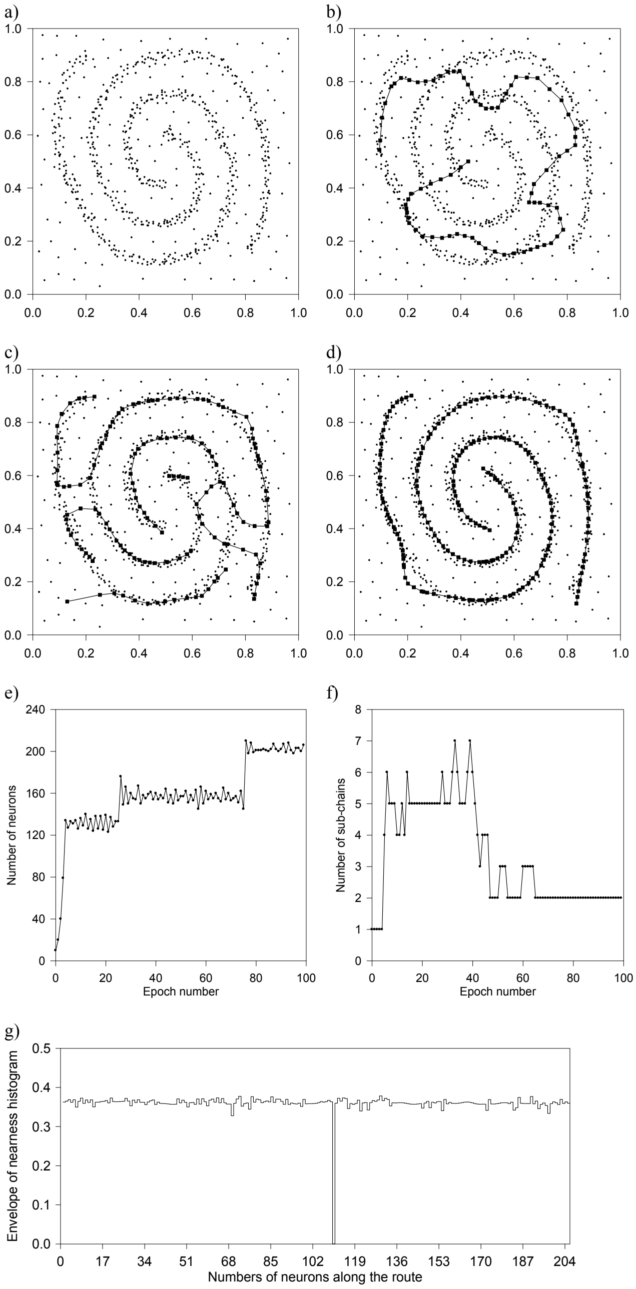

Fig. 1. Two-spiral data set (a) and the “route” in it in learning epochs: b) no. 3, c) no. 20, and d) no. 100 (end of learning), as well as the plots of the number of neurons (e) and the number of sub-chains (f) vs. epoch number, and the envelope of nearness histogram for the final route of Fig. 1d (g)

Rys. 1. Zbiór danych typu dwóch spiral (a) oraz wyznaczona w nim “droga” w epokach uczenia: b) nr 3, c) nr 20 i d) nr 100 (zakończenie uczenia), jak również wykresy liczby neuronów (e) i liczby podłańcuchów (f) w kolejnych epokach uczenia oraz obwiednia histogramu bliskości dla końcowej drogi z rys. 1d (g)

As the learning progresses, the dynamic neural system modifies the route (represented by the neuron chain) in the data set (see Figs. 1b, 1c, and 1d for epoch nos. 3, 20, and 100 (end of learning)) to finally detect two clusters. In order to achieve this goal the system:

a) automatically adjusts the number of neurons in the network (starting from arbitrarily chosen 10 neurons and finally stabilizing on 206 neurons),

b) automatically adjusts the number of sub-chains of the network (equal to the number of clusters) finally stabilizing on 2 sub-chains.

As it can be seen in Fig. 1g (the envelope of nearness histogram for the route of Fig. 1d), the proposed clustering technique provides correct, clear and easily-verified image of the cluster distribution in the problem under consideration. The plot of Fig. 1g confirms the occurrence of two clusters with distinct boundary between them (indicated by minimum on that plot).

3. Application to clustering of complex time series measurement data

The dynamic self-organizing neural network will now be applied to the clustering of complex and multidimensional measurement-type data set (the so-called *Synthetic Control Chart Time Series*) available at WWW server of the Department of Information and Computer Science, the University of California at Irvine [7].

The number of classes (equal to 6: *Normal*, *Cyclic*, *Increasing trend*, *Decreasing trend*, *Upward shift* and *Downward shift*) and the class assignments are known here, which allows us for direct verification of the results obtained. Obviously, the knowledge about the class assignments by no means will be used by the clustering system (it works in a fully unsupervised way). The data set contains 600 records (100 records for each of six classes); each record is described by as many as 60 attributes (data samples, that is, numerical values of a given parameter over time). Fig. 2 shows the plots of exemplary time series (ten records from each class) included in the data set.

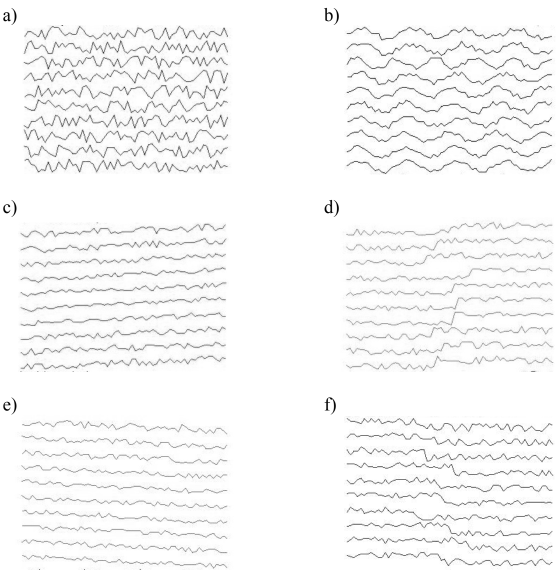

Fig. 2. The plots of exemplary time series included in the data set (ten records from each class: *Normal* (a), *Cyclic* (b), *Increasing Trend* (c), *Upward Shift* (d), *Decreasing Trend* (e), *Downward Shift* (f))

Rys. 2. Przykładowe przebiegi czasowe zawarte w zbiorze danych (dziesięć przebiegów z każdej klasy: *Normal* (a), *Cyclic* (b), *Increasing Trend* (c), *Upward Shift* (d), *Decreasing Trend* (e), *Downward Shift* (f))

Figs. 3 and 4 present the performance of the proposed clustering technique in the considered data set. As the learning progresses, the neural system adjusts the overall number of neurons in its network (Fig. 3a) that finally is equal to 101 as well as the number of sub-chains (Fig. 3b) finally achieving the value equal to 6. The number of sub-chains is equal to the number of clusters detected in a given data set.

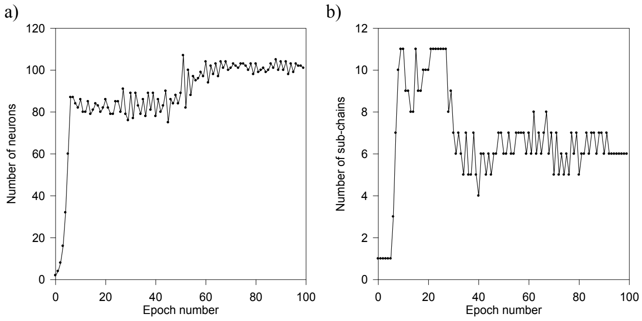

Fig. 3. The plots of the number of neurons (a) and the number of sub-chains (b) vs. epoch number for the *Synthetic Control Chart Time Series* data set

Rys. 3. Wykresy liczby neuronów (a) i liczby podłańcuchów (b) w kolejnych epokach uczenia dla zbioru danych *Synthetic Control Chart Time Series*

The envelope of the nearness histogram for the route in the attribute space of the considered data set (Fig. 4) reveals perfectly clear image of the cluster distribution in it, including the number of clusters and the cluster boundaries (indicated by 5 local minima on the plot of Fig. 4). After performing the so-called calibration of the final neural network, class labels can be assigned to particular sub-chains of the network as shown in Fig. 4. The neuron sub-chain representing the cluster *Upward shift* contains neurons nos. 1 to 21, the cluster *Cyclic* – neurons nos. 22 – 37, the cluster *Decreasing trend* – neurons nos. 38 – 62, the cluster *Increasing trend* – neurons nos. 63 – 76, the cluster *Normal* – neurons nos. 77 – 94, and the cluster *Downward shift* – neurons nos. 95 – 101.

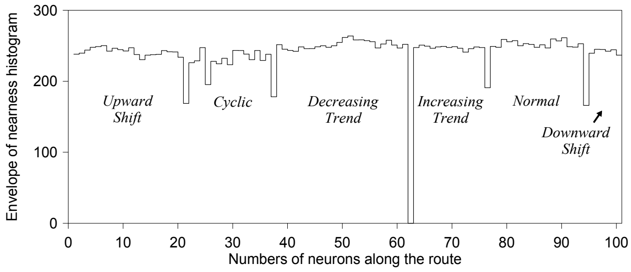

Fig. 4. The envelope of nearness histogram for the final route (final neuron chain) in the attribute space of the *Synthetic Control Chart Time Series* data set

Rys. 4. Obwiednia histogramu bliskości dla końcowej drogi (końcowego łańcucha neuronów) w przestrzeni atrybutów zbioru danych *Synthetic Control Chart Time Series*

Since the number of classes and the class assignments are known in the original data set, direct verification of the obtained result is also possible. Numerical results of clustering have been collected in Table 1.

Tab. 1. Clustering results of the *Synthetic Control Chart Time Series* data set

Tab. 1. Wyniki grupowania zbioru danych *Synthetic Control Chart Time Series*

Class label	Number of decisions for sub-chain labelled$^{(*)}$:						Number of correct decisions	Number of wrong decisions	Percentage of correct decisions
	N	*C*	*It*	*Us*	*Dt*	*Ds*			
Normal	100	0	0	0	0	0	100	0	100%
Cyclic	0	100	0	0	0	0	100	0	100%
Increasing trend	0	0	74	26	0	0	74	26	74%
Upward shift	0	0	0	100	0	0	100	0	100%
Decreasing trend	0	0	0	0	100	0	100	0	100%
Downward shift	0	0	0	0	58	42	42	58	42%
ALL	100	100	74	126	158	42	516	84	86%

(*) *N=Normal, C=Cyclic, It=Increasing trend, Us=Upward shift, Dt=Decreasing trend, Ds=Downward shift*

The average percentage of correct decisions, equal to 86%, regarding the class assignments is very high, especially that it has been achieved by the unsupervised-learning system (working without any prior information on the number of clusters in the data set) and operating on complex and multidimensional data set. Moreover, it is worth emphasizing that the system provides 100%(!) correct decisions for four out of six classes. Only some data of *Increasing trend* class are misclassified as *Upward shift* data and some data of *Downward shift* class are misclassified as *Decreasing trend* data. However, the plots in both pairs of trends are very similar to each other (see Figs. 2cd and Figs. 2ef) and distinguishing them may pose a challenge even for humans.

4. Signal filtering by means of dynamic self-organizing neural networks – an outline

After deactivation of the mechanisms governing the disconnection of the neuron chain and the reconnection of some of the sub-chains again (that is, the operations nos. 2, 3 and 5) the dynamic self-organizing neural networks become an efficient tool for signal filtering. Moreover, by regulating the remaining mechanisms governing the removal of low-active neurons and the insertion of new neurons, that is, the mechanisms determining the length of the neuron chain it is also possible to regulate the “level” of signal filtering.

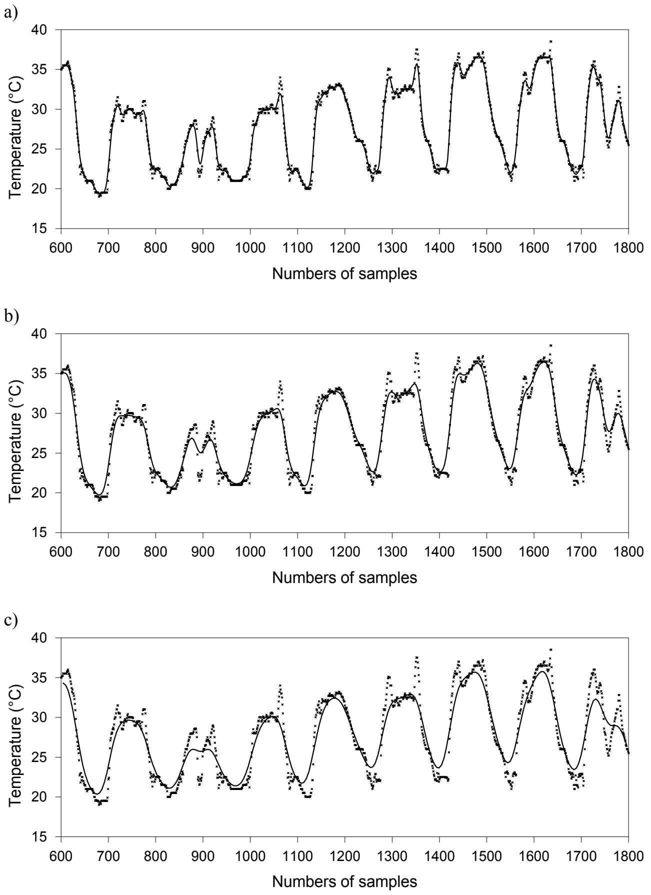

Fig. 5. *Equiptemp* data set (samples from no. 600 to no. 1830) before filtering (dotted line) and after filtering (continuous line) for the number of neurons in the network equal to 2595 (a), 1322 (b), and 849 (c)

Rys. 5. Dane *Equiptemp* (próbki od nr 600 do nr 1830) przed filtracją (linia przerywana) i po filtracji (linia ciągła) dla liczby neuronów sieci równej 2595 (a), 1322 (b) oraz 849 (c)

The test of the dynamic self-organizing neural networks as signal filtering tools will be briefly presented with the use of *Equiptemp* data set, available in *Time Series Library* [6]. Data set contains 4325 measurement samples – offset equipment temperature (degrees Celsius, equipment used for radioactive measurement) observed every 10 minutes for one month (July 2006). Fig. 5 presents – for better readability – only the samples from no. 600 to no. 1830 before filtering (dotted line) and after filtering (solid line) for different setting of the mechanisms governing the length of the neuron chain. It can be seen that the longer the neuron chain (the closer the number of neurons in the chain to the number of data samples) the lower “level” of signal filtering).

5. Conclusions

The applications of the dynamic self-organizing neural networks with one-dimensional neighbourhood (introduced by the same authors in [3, 4, 5]) to clustering of complex, multidimensional measurement-type data sets and – after some modifications – to signal filtering have been presented in this paper. The proposed networks in the course of learning are able to disconnect their neuron chains into sub-chains and to reconnect some of the sub-chains again as well as to dynamically adjust the overall number of neurons in the system. These features enable them to fit in the best way the structures “encoded” in data sets without any prior information regarding the data. The proposed technique has been applied to clustering of complex and multidimensional measurement-type data set (the so-called *Synthetic Control Chart Time Series*) available at WWW server of the Department of Information and Computer Science, the University of California at Irvine [7]. Moreover, the filtering of *Equiptemp* measurement data set available from *Time Series Library* [6] by means of the proposed networks has also been briefly presented.

Taking into account the results that have been reported in this paper, it is clear that the proposed dynamic self-organizing neural networks are powerful tools for clustering of complex, multidimensional measurement-type data sets as well as for signal filtering.

6. References

[1] Gorzałczany M.B., Rudziński F.: Applicacion of genetic algorithms and Kohonen networks to cluster analysis, in L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2004, Lecture Notes in Artificial Intelligence 3070, Springer-Verlag, Berlin, Heidelberg, New York, 2004, pp. 556-561.

[2] Gorzałczany M.B., Rudziński F.: Modified Kohonen networks for complex cluster-analysis problems, in L. Rutkowski, J. Siekmann, R. Tadeusiewicz, L.A. Zadeh (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2004, Lecture Notes in Artificial Intelligence 3070, Springer-Verlag, Berlin, Heidelberg, New York, 2004, pp. 562-567.

[3] Gorzałczany M.B., Rudziński F.: Cluster Analysis via Dynamic Self-organizing Neural Networks, in L. Rutkowski, R. Tadeusiewicz, L.A. Zadeh, J. Zurada (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2006, Lecture Notes in Artificial Intelligence 4029, Springer-Verlag, Berlin, Heidelberg, 2006, pp. 593-602.

[4] Gorzałczany M.B., Rudziński F.: Application of dynamic self-organizing neural networks to WWW-document clustering, International Journal of Information Technology and Intelligent Computing, Vol. 1, No. 1, 2006, pp. 89-101.

[5] Gorzałczany M.B., Rudziński F.: WWW-newsgroup-document clustering by means of dynamic self-organizing neural networks, in L. Rutkowski, R. Tadeusiewicz, L.A. Zadeh, J. Zurada (Eds.), Artificial Intelligence and Soft Computing – ICAISC 2008, Lecture Notes in Artificial Intelligence 5097, Springer-Verlag, Berlin, Heidelberg, 2008, pp. 88-96.

[6] Hyndman, R.J. (n.d.): Time Series Data Library, accessed on July 10, 2010 (http://robjhyndman.com/TSDL).

[7] UCI Knowledge Discovery in Databases Archive of the Department of Information and Computer Science, University of California at Irvine (http://kdd.ics.uci.edu).

otrzymano / received: 25.09.2010
przyjęto do druku / accepted: 01.11.2010 *artykuł recenzowany*