

A New Method for Adaptive Selection of Self-Organizing Map Self-Training Endpoint

Liudas STAŠIONIS, Artūras SERACKIS

Vilnius Gediminas Technical University, Saultekio al. 11, Vilnius

{liudas.stasionis, arturas.serackis}@vgtu.lt

Abstract. The paper presents a new method for adaptive selection of Self-Organizing Map (SOM) self-training endpoint. A method is based on the estimation of the newly introduced parameter Δ_{init} and the learning depth parameter κ . In order to propose an optimal range of κ values, the influence of the selected learning depth parameter to the performance of SOM was tested experimentally using input data with uniform distribution. Additionally, four endpoint selection approaches were tested in spectrum sensing application where the SOM based detector was used to detect primary user emissions in 25 MHz wide spectrum band. Three alternative SOM self-training endpoint selection methods were tested on the same topology based SOM. In comparison to SOM self-training endpoint selection algorithm, based on the cluster quality estimation, the proposed method required from 2.6% (for the SOM with small number of neurons) to 44.6% (for the SOM with higher number of neurons) less iterations to reach the endpoint and preserve the similar sensitivity of the spectrum sensor based on SOM.

Keywords: Self-Organizing Map, SOM self-training, adaptive endpoint, spectrum sensor

1 Introduction

High dimensional data analysis is a challenging task. The Self-Organizing Map (SOM) is frequently used for high dimensional data clustering and dimensionality reduction (Bernataviien et al. 2006; Bernataviien et al. 2007). In addition, it has already been shown that the application of Self-Organizing Feature Map could be applied as a part of the spectrum sensor in cognitive radio systems (Baban et al. 2013; Liu et al. 2015). The task of the spectrum sensor is to locate gaps in the analyzed radio frequency range in order to efficiently apply dynamic spectrum management techniques (Yucek and Arslan 2009). During analysis of a wide spectrum band, it is possible to find different types of the primary user signal transmissions with unknown frequency characteristics, signal modulation type, signal strength, etc. A Self-Organizing Map is used to cluster the signal spectrogram features into clusters that indicates unknown primary user signal transmissions and clusters, indicating the noise – a channel, free for transmission of a secondary user signal.

All signal processing steps in the designed spectrum sensor should be made in real-time. The SOM self-training should be done in real time and the self-training algorithm should be adjusted for hardware-based implementation. However, the number of iterations needed for SOM convergence is related to the number of neurons in the SOM structure. The convergence of SOM, constructed from N nodes, is usually obtained after $N \times 500$ iterations. In addition, the long self-training procedure may cause the over-fitting of the SOM (Weijters et al. 1997; Lawrence et al. 1997; Lampinen and Kostiaainen 1999; Haese and Goodhill 2001; Lampinen and Kostiaainen 2000). Therefore, the optimization of the SOM self-training procedure should be made in order to design a SOM based spectrum sensor for practical application.

New SOM topology for real-time spectrum sensing and fast convergence was proposed in previous author's work (Staionis and Serackis 2015). The experimental investigation has shown that the performance of the spectrum sensor with the proposed topology remains as high as the performance of the spectrum sensor based on a hexagonal topology. For the specific radio spectrum, the sensor performance using the proposed topology was superior compared to spectrum sensors based on hexagonal, grid or rhombus SOM topologies. However, this newly proposed SOM topology was very sensitive to the number of iterations used for self-training. The desired SOM topology was achieved very quickly, but after several more iterations, the over-fitting of the SOM was obtained. An appropriate method was needed to select an endpoint during SOM self-training in convergence point.

The convergence of SOM depends on three main aspects: selection of initial weights, learning rate and neighborhood size. Therefore, limiting the SOM self-training process by maximum number of iterations is not reasonable, because the SOM self-training performance may be not very sensitive to the number of self-training steps (Bogdan et al. 2008). The selection of the SOM self-training endpoint can be made by monitoring the estimated mean value of the cost function (Kohonen 1991; Lampinen and Oja 1992). As an alternative, the tracking of the $\eta(l) \simeq \eta(l - 1)$ can be used as indication to stop the self-training, if the changes of the learning rate becomes insignificant (Vegas-Azcarate et al. 2005). The third alternative method for endpoint selection is based on the SOM cluster quality measure (Herbert and Yao 2007).

The preliminary experimental tests of the currently available SOM self-training endpoint selection methods showed that the automatic selection of the endpoint was made too early (the sensitivity of the SOM based spectrum sensor could be improved by adding additional self-training iterations) or too late (the sensitivity of the SOM based spectrum sensor did not change).

In this paper, we propose an alternative method for adaptive endpoint selection during the self-training of SOM. A method is based on estimation of the newly introduced parameter Δ_{init} and the learning depth parameter κ .

While the method for adaptive endpoint selection was designed to be used in SOM based spectrum sensor, an additional investigation was performed in order to ensure that the selected approach is suitable for classical SOM self-training. Additionally, an experimental investigation was performed in order to propose an optimal range of κ values and to test the performance of the proposed method in spectrum sensing applications.

To compare the results of our proposed break point selection method to alternatives, three alternative SOM self-training endpoint selection methods were tested on the same topology based SOM. The results of experimental investigation and performed tests has proved the better performance of the proposed endpoint selection method in comparison to the alternative ones.

2 Self-training Endpoint Selection Methods for SOM

There are two approaches available for SOM training: supervised SOM (Pateritsas et al. 2004) and un-supervised SOM (Fritzke 1994). The supervised SOM training requires additional external data during SOM training procedure and was not analyzed in this paper.

During un-supervised SOM training, the network status or the changes of node weights are monitored. In addition, the representation of input data by SOM can be analyzed. The changes of SOM node weights reflects the changes in map structure. If the weights changes are negligible during self-training, the SOM self-training process is terminated. However, negligible changes of the SOM node weighs during self-training does not guarantee that the obtained topology properly represents input data.

The endpoint selection during self-training of SOM requires continuous monitoring of the self-training process. It is possible to classify currently proposed approaches into two types: continuous monitoring of the current SOM structure during self-training process or analysis of the input data representation by the SOM.

Using the SOM structure monitoring approach, the SOM self-training process is suspended, when the changes of neuron weights become insignificant (Vegas-Azcarate et al. 2005). The weight update of the winner neuron Δw highly depends on the selected adaptive learning rate η according to the following expression:

$$w_{ij}(n+1) = w_{ij}(n) + \eta(n) \|\mathbf{In}(n) - \mathbf{w}_i(n)\|. \quad (1)$$

The learning rate η changes adaptively and is being estimated accordingly to the following expression:

$$\eta(n) = \eta_0 e^{-\frac{n}{\tau}}, \quad (2)$$

here η_0 is the initial value of the learning rate.

During SOM self-training process, the η decreases exponentially. If the learning rate changes are insignificant and $\eta(l) \simeq \eta(l-1)$, the further update of the neuron weights is not reasonable. However, low value of the learning rate η does not mean that the SOM represents input data well.

In order to monitor the input data representation by the SOM, distance changes between the input \mathbf{In} and the winner neuron weight vector \mathbf{w}_i should be estimated (Mora and Fiesler 1995; Solodov and Svaite 2000; Hulle 2000). During such approach, the endpoint is initiated when the distance between inputs and winner neuron weights reach their minimum $I(\mathbf{In})$:

$$I(\mathbf{In}) = \arg \min_{ij} \|\mathbf{In} - \mathbf{w}_i\|. \quad (3)$$

As an alternative to the estimation of the distance between input and neuron weights, the quality of the input data clustering by a set of neurons may be used. However, these alternative approaches require performing additional analysis using the higher order statistical data (Kayacik et al. 2007; Herbert and Yao 2007).

3 Investigation of the New Method for Self-training Endpoint Selection

The SOM self-training endpoint selection method, proposed in this paper, was based on the idea of monitoring the input data representation by the SOM. The estimated value of $I(\mathbf{In})$ was compared to the threshold θ_{end} , used to initiate the endpoint.

In order to select an appropriate threshold θ_{end} for the endpoint, the attention should be drawn to the first cycles of the SOM self-training (see Fig. 1, a). From the Fig. 1 it was clear that the signal, obtained by continuously estimating $I(\mathbf{In})$, had a spiking nature. However, during SOM self-training the magnitude of the spikes decreased. The monotonic decrease of the distance between input and SOM neuron weights can be seen also in the characteristics of the averaged $I(\mathbf{In})$. In the given example, the maximum of $I(\mathbf{In})$ was achieved at the 16th self-training iteration. The current input-to-weights distance comparison with the maximum, achieved in the first cycles of SOM self-training, was used to make the final decision in the threshold θ_{end} selection phase.

In order to get the more clear view of the $I(\mathbf{In})$ dynamics, the high-pass filter was applied. The result after differentiation, given in Fig. 1, b, showed, that the greater input-to-weights distance invoked significant changes of neuron weights (see equation 3 and 1).

The signal in Fig. 1, c, was obtained by estimating the square of the differentiation result, given in Fig. 1, b. From the given example, it was seen that the weight changes of the SOM neurons were decreased during the period of 80 self-training iterations. A new variable Δ_{init} was introduced for adaptive endpoint estimation. The newly introduced estimate Δ_{init} showed the average level of SOM weight changes during self-training process. Δ_{init} was calculated according to the following mathematical expression:

$$\Delta_{\text{init}} = \sqrt{\frac{1}{N_f} \sum_{n=0}^{N_f-1} \left(I(\mathbf{In}(n+1)) - I(\mathbf{In}(n)) \right)^2}, \quad (4)$$

here N_f is the number of $I(\mathbf{In}(n))$ estimates in the analysis frame.

The threshold θ_{end} was adaptively changed together with estimated Δ_{init} value, according to the following expression:

$$\theta_{\text{end}} = \kappa \Delta_{\text{init}} = \kappa \sqrt{\frac{1}{N_f} \sum_{n=0}^{N_f-1} \left(I(\mathbf{In}(n+1)) - I(\mathbf{In}(n)) \right)^2}, \quad (5)$$

here κ is the additional parameter in the range $(0, 1]$, introduced in order to control the depth of SOM self-training.

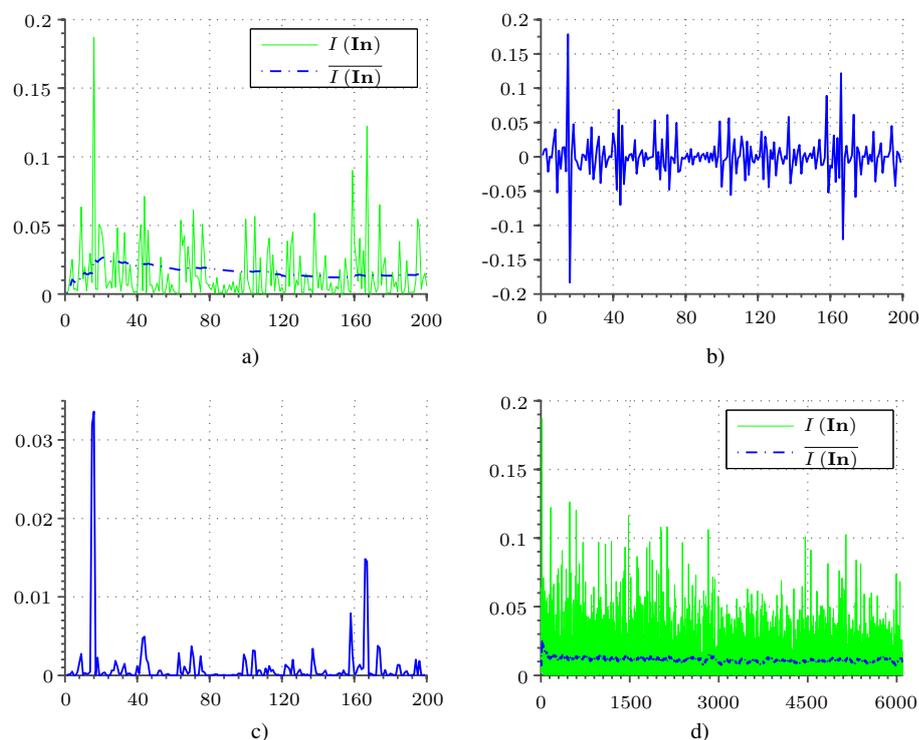


Fig. 1. Illustration of input-to-weights distance estimate $I(\mathbf{In})$ changes during SOM self-training: a – $I(\mathbf{In})$ continuous estimate and it's average $\overline{I(\mathbf{In})}$ over 200 iterations; b – $I(\mathbf{In})$ continuous estimate after differentiation; c – squared differential signal; d – $I(\mathbf{In})$ continuous estimate and it's average $\overline{I(\mathbf{In})}$ over 6200 iterations

The $I(\mathbf{In}(n))$ estimate changed dynamically in time with a high frequency. In order to make a decision about SOM self-training endpoint initialization, the threshold was compared to the average estimate $\overline{I(\mathbf{In})}$. The illustration of the $\overline{I(\mathbf{In})}$ dependences are given in Fig. 1, a and Fig. 1, d.

4 Selection of Optimal Values for Self-training Endpoint

An experimental investigation was performed in order to measure the influence of each selected parameter (N_f, κ) and to select the optimal values for practical application of proposed endpoint selection method, based on the average level of SOM weight changes (ALWCh).

4.1 SOM Training Performance Dependences on the Analysis Frame Size

In order to investigate the dependences of SOM self-training performance and Δ_{init} to the selected analysis frame size N_f , four SOM, different in size (2×2 , 4×4 , 6×6 and

8×8 neurons) were trained. The SOM self-training results for the first 500 iterations are illustrated in Fig. 2.

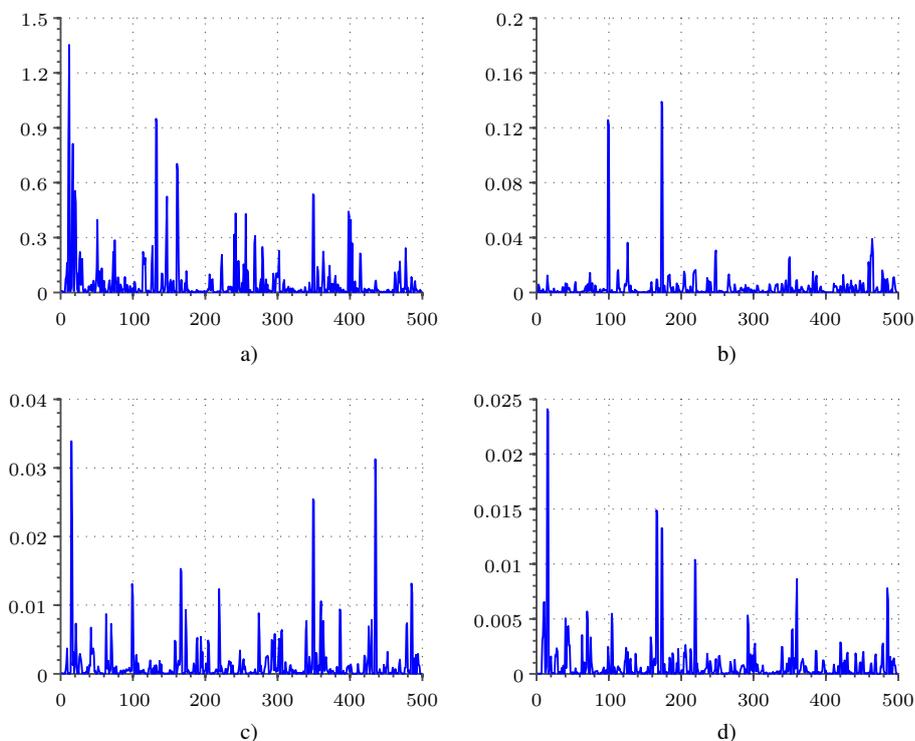


Fig. 2. Squared $I(\mathbf{In})$ signal after differentiation for different SOM size: a – 2×2 ; b – 4×4 ; c – 6×6 ; d – 8×8

The Δ_{init} was calculated for different analysis frame width N_f in order to find the optimal value for the most of SOM structures. During investigation, the parameter N_f was changed from 25 to 300 with the step size of 25. The results are summarized in Table 1.

During the experiments with the 2×2 SOM structure, the Δ_{init} has reached its maximum with $N_f = 25$. The analysis of experimental investigation results showed that the group of spikes between 100th and 200th iterations should be included into the estimation of Δ_{init} because the changes of SOM weights still remains comparatively high. In addition, the Δ_{init} estimate was reduced by 36%, when the frame width was increased from $N_f = 25$ to $N_f = 300$.

Using the 4×4 SOM structure, the main spikes were achieved at 99th and 178th iterations. Therefore the frame width of $N_f = 200$ iterations showed the maximum value of $\Delta_{init} = 0.069$.

Table 1. Δ_{init} estimates for various frame width

SOM size	Frame width N_f							
	25	50	75	100	150	200	250	300
2×2	<u>0.469</u>	0.359	0.324	0.296	0.297	0.277	0.263	0.258
4×4	0.040	0.039	0.042	0.054	0.06	<u>0.069</u>	0.067	0.064
6×6	<u>0.057</u>	0.048	0.043	0.04	0.036	0.038	0.036	0.036
8×8	<u>0.054</u>	0.045	0.04	0.036	0.032	0.035	0.033	0.032

The main spike for the 6×6 SOM structure was obtained at 15th self-training iteration. However, the spikes at 350th and 435th iterations had similar magnitude. In order to cover all spikes with a very high magnitude, the frame width should be selected equal to $N_f = 450$. The value $\Delta_{\text{init}} = 0.038$ is less than observed for smaller SOM structure for the same frame width. However the effect of nonlinear decrease of Δ_{init} , increasing the size of SOM structures, was observed for all frame width above $N_f = 100$ (Table 1).

An experimental investigation performed on the 8×8 size SOM structure showed that the increase of the SOM size reduces the fluctuations of the Δ_{init} estimate in the analyzed frame width range. The stability of Δ_{init} estimate is achieved after 150–200 SOM self-training iterations with fluctuations in the range of 10%–13%.

4.2 SOM Training Performance Dependences on a Learning Depth

In order to investigate the influence of the learning depth parameter κ to the SOM self-training performance the 8×8 size SOM topology was trained with four different values of κ . Two random signals with uniform distribution were used as inputs during the experimental investigation. During the experimental test, we have obtained different number of passed iterations before the endpoint was activated (see Fig. 3).

The learning depth κ was changed in the range $[0.15, 0.3]$. The shortest SOM learning cycle was achieved with $\kappa = 0.3$. The endpoint at $\theta_{\text{end}} = 0.01065$ was obtained at 811th iteration. It is seen from Fig. 3, a, that the SOM did not reach the regular structure. However, the neurons at the right side were already starting to adjust their weights in the right order.

The SOM learning endpoint with $\kappa = 0.25$ was activated after 3120 iterations. The threshold $\theta_{\text{end}} = 0.008875$ obtained with $\kappa = 0.3$. The results of experimental tests are illustrated in Fig. 3, b. The obtained SOM structure was more regular comparing to the results shown in Fig. 3, a, however the SOM weights still needed additional adjustment in order to represent the uniform noise distribution.

When the learning depth was decreased to $\kappa = 0.2$, the endpoint was activated after 6146 iterations and SOM neurons formed a regular structure divided into four groups of neurons (see Fig. 3, c). The threshold θ_{end} , at which the endpoint was activated, also had a lower value, equal to 0.0071.

An additional decrease of learning depth to $\kappa = 0.15$ did not show the significant changes of the SOM structure after the 6146 iterations has passed. The self-training

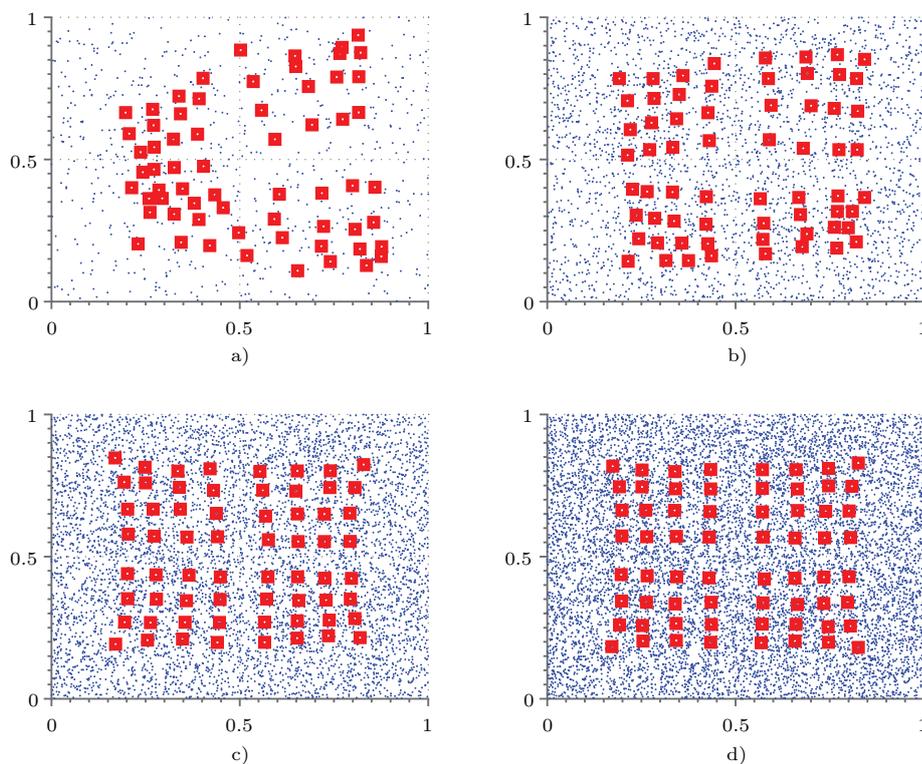


Fig. 3. SOM structure after self-training: a – 8×8 SOM structure after 811 iterations with $\kappa = 0.3$; b – 8×8 SOM structure after 3120 iterations with $\kappa = 0.25$; c – 8×8 SOM structure after 6146 iterations with $\kappa = 0.2$; d – 8×8 SOM structure after 10 000 iterations with $\kappa = 0.15$

was stopped because of the reached maximum allowed number of iterations (equal to 100 000) in the algorithm and the endpoint was not activated. During the experimental investigation, the regular structure of the SOM has been achieved with $\kappa = 0.2$. However, the other type of input signal (not random) may require the higher value of κ . Few additional tests performed with the signals having different statistical distributions showed, that the κ should be chosen in the range from 0.2 to 0.25.

5 Comparison of Different SOM Self-training Endpoint Selection Methods

ALWCh endpoint selection method proposed in this paper works in an un-supervised manner. Therefore, three alternative methods for SOM self-training endpoint selection were used to compare the performance of the proposed approach: a method based on the mean value of the cost function (MVCF); a method based on monitoring learning rate changes (LRCh) and a method based on cluster quality estimation (CQE). The alternative approaches were selected by taking into account the additional computational cost

during the application of the selected method in real-time. The amount of additional calculations, which were needed to make a decision about the self-training endpoint, had to be less or equal to the proposed method. The results, obtained by using four different methods on the same SOM of size 8 are shown in Fig. 4.

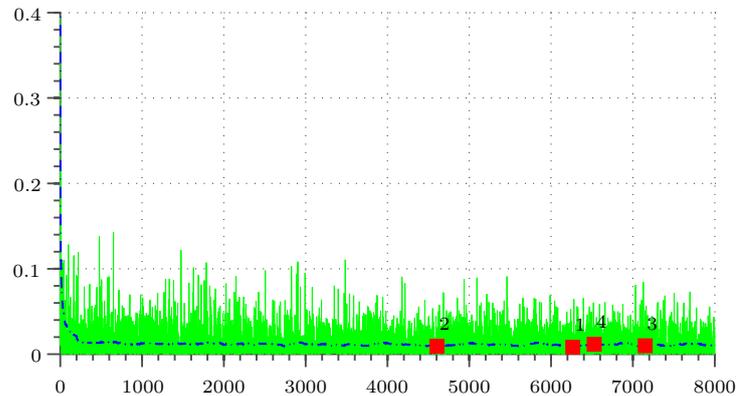


Fig. 4. Illustration of $I(\text{In})$ changes at every self-training iteration with marked self-training endpoints, obtained during experimental investigation

The self-training endpoint, estimated by MVCF method is shown as **mark 2** in Fig. 4. The complexity of this endpoint activation method is similar to the complexity of the ALWCh method proposed in this paper. The additional calculations, needed for endpoint activation parameter are performed at the same time.

The self-training endpoint, estimated by LRCh method is shown as **mark 3** in Fig. 4. The method tracks the dynamics of SOM weight changes. If the changes of the SOM weights becomes insignificant, the endpoint is activated and the self-training is suspended. Because no additional calculations are performed during the SOM self-training process, the implementation complexity does not increase.

The self-training endpoint, estimated by CQE method is shown as **mark 4** in Fig. 4. The cluster quality, measured in this method, shows how well the SOM classify the inputs and how well each neuron represents (as a cluster) it's input data. The rating was performed in each training cycle. Therefore, this method was the most computationally intensive in our test. The endpoint was activated with a delay, because the quality of current SOM in this method was verified for various neurons and with various input vectors.

An activation of 8×8 SOM self-training endpoint accordingly to the proposed ALWCh method was done at 6265th iteration shown in Fig. 4 as **mark 1**. The number of iterations passed from the beginning of the SOM self-training was above 6000. From the Fig. 3, c, it is clear that the SOM at the point **mark 1** was already well trained.

The CQE endpoint activation method passed 6527 iterations before activation (**mark 4** in Fig. 4). It took 262 (4.19%) more iterations comparing to the proposed ALWCh

method. The SOM weight changes were insignificant in 7151 iteration accordingly to the least computationally intensive LRCh method (**mark 3** in Fig. 4). The MVCF endpoint activation method gave unacceptable results because the self-training was stopped too early, at the 4606 iteration. The SOM structure after 4606 iterations was still distorted and needed additional adjustments.

The results given in Table 2 presents the comparison of the different endpoint activation methods for the smaller SOM topologies. From these results, it was clear that all endpoint activation methods preserves the same tendencies during SOM self-training. The CQE method had near 250 iterations delay in comparison to proposed ALWCh method. The activation of the endpoint by MVCF method was always earliest. However, the visual analysis of the SOM weight adjustment showed that the activation of the endpoint according to MVCF method was too early on experimental data and should not be used.

Table 2. Comparison of the endpoint activations for the SOM of the different size

	2×2	3×3	4×4	5×5	6×6	7×7
MVCF	-25%	-8.4%	-4.1%	-9.1%	4.4%	-31.8%
LRCh	38.9%	24.5%	31.1%	18.9%	33.8%	14.2%
CQE	15.3%	24.5%	31.1%	5.1%	22.5%	3.4%
ALWCh	1627(0%)	2868(0%)	3405(0%)	4622(0%)	4276(0%)	6194(0%)

5.1 Application of SOM for Spectrum Sensing

A set of additional experiments were performed by using SOM based spectrum sensor. The self-training of SOM was performed in real time, additionally in parallel analyzing the current state of SOM self-training results.

25 MHz wide radio spectrum band at 928 MHz frequency was analyzed during experimental investigation. The recorded radio signals were pre-analyzed by cyclostationary feature estimation based spectrum sensor to detect and annotate the primary user signal emissions. A SOM based spectrums sensor was applied to the same radio signals. As a result, the rate of detected primary user emissions P_D and the rate of spectrum sensor false alarms P_{FA} were collected for each SOM based detector with different self-training endpoint setting approach. The number of self-training iterations passed before the endpoint was activated is given in Table 3.

The activation of the SOM self-training endpoint is made according to the training state analysis results, estimated in parallel to self-training procedure. Therefore, the computationally intensive analysis algorithm activates the endpoint with a delay. Such situation was observed for the endpoint selection method based on the analysis of cluster quality. Both, the proposed ALWCh method and CQE methods showed similar performance in primary user emission detection (see Table 4). However, three from eight SOM detectors, which used CQE method during self-training, failed to detect all primary user emissions in the radio signal spectrogram.

Table 3. Comparison of the SOM self-training endpoint activation points during spectrum sensing application

SOM size	ALWCh method	CQE method	LRCh method	MVCF method
3 × 3	1295	1329	1901	492
4 × 4	1324	1391	2852	650
4 × 5	1454	1567	3804	1031
5 × 5	1373	1545	4753	789
5 × 6	1258	1524	5700	876
6 × 6	<u>1115</u>	1612	6632	<u>1422</u>
5 × 8	1339	1866	7825	857
5 × 9	2262	2346	8742	907

Table 4. Comparison of the SOM based spectrum sensor performance with different endpoint selection algorithms

SOM size	ALWCh method		CQE method		LRCh method		MVCF method	
	P_D	P_{FA}	P_D	P_{FA}	P_D	P_{FA}	P_D	P_{FA}
3 × 3	1	0.0169	1	0.0183	1	0.0159	0.897	0
4 × 4	<u>1</u>	<u>0.0156</u>	0.9985	0.0057	1	0.0133	0.859	0
4 × 5	0.9801	0	<u>1</u>	<u>0.0142</u>	1	0.0115	0.931	0
5 × 5	1	0.0202	1	0.0196	1	0.0121	0.884	0
5 × 6	1	0.0165	0.995	0.0089	1	0.0125	0.894	0
6 × 6	1	0.0175	1	0.0214	<u>1</u>	<u>0.0108</u>	<u>1</u>	<u>0.0175</u>
5 × 8	1	0.0242	0.99	0.0126	0.995	0.0101	0.91	0
5 × 9	1	0.0168	1	0.0175	0.98	0.0089	0.923	0

A good sensitivity was achieved during application of SOM based spectrum sensor with self-training endpoint selection by LRCh method. The false alarm rate P_{FA} was lower comparing to all methods which have found all primary user emissions (see Table 4). However, the number of iterations that passed before the endpoint was activated was much higher comparing to alternative methods. Especially this difference was high when the number of SOM neurons was increased (see Table 3). The high number of SOM self-training iterations means that it takes more time for the SOM based spectrum sensor in order to adapt to the dynamic radio environment and to start the primary user emission detection tasks.

6 Conclusions

The endpoint activation method proposed in this paper allowed us to reduce the number of SOM self-training iterations before the SOM based spectrum sensor can be applied for primary user emission detection.

The proposed method was compared to alternative approaches, suitable for implementation in real-time embedded system. The performance of the proposed method was slightly better in comparison to the alternative methods according to two parameters: emission detection rate and number of iterations before the endpoint is initiated. Only the sensor with SOM of 4×5 size (from eight tested in total) was not able to detect all primary user emissions, while alternative methods reduced the sensitivity of two or more SOM based sensors.

In comparison to SOM self-training endpoint selection algorithm, based on the cluster quality estimation, the proposed method required from 2.6% (for the SOM with small number of neurons) to 44.6% (for the SOM with higher number of neurons) less iterations to reach the endpoint and preserve the similar sensitivity of the spectrum sensor.

References

- Baban, Sh., Holland, O. and Aghvami, H. (2013) Wireless Standard Classification in Cognitive Radio Networks Using Self-Organizing Maps. In *Proceedings of the Tenth International Symposium on Wireless Communication Systems (ISWCS 2013)*, 1–5.
- Bernataviiien, J., Dzemyda, G., Kurasova, O. and Marcinkevicius, V. (2006). Optimal decisions in combining the som with nonlinear projection methods. *European Journal of Operational Research*, **173**, 729–745.
- Bernataviiien, J., Dzemyda, G., Kurasova, O., Marcinkevicius, V. and Medvedev, V. (2007). The problem of visual analysis of multidimensional medical data. *Models and Algorithms for Global Optimization*, Springer US, 277–298.
- Brugger, D, Bogdan, M. and Rosenstiel, W. (2008). Automatic cluster detection in Kohonen's SOM. *IEEE Transactions on Neural Networks*, **3(19)**, 442–459.
- Fritzke, B. (1994). Growing cell structures a self-organizing network for unsupervised and supervised learning. *Neural networks*, **7**, 1441–1460.
- Haese, K. and Goodhill, G. J. (2001). Auto-som: recursive parameter estimation for guidance of self-organizing feature maps. *Neural computation*, **13**, 595–619.
- Herbert, J. and Yao, J. (2007). Gtsom: Game theoretic self-organizing maps. In K. Chen, & L. Wang (Eds.), *Trends in Neural Computation*. Springer Berlin Heidelberg **35** of *Studies in Computational Intelligence*, 199–223.
- Hulle, M. M. V. (2000). *Faithful Representations and Topographic Maps: From Distortion- to Information-Based Self-Organization*. New York, NY, USA: John Wiley & Sons, Inc.
- Vegas-Azcarate, S., Gautama, T. and Van Hulle, M. (2005). Topology preservation in topographic maps. *Technical Reports on Statistics and Decision Sciences, Rey Juan Carlos University*.
- Kayacik, H. G., Zincir-Heywood, A. N. and Heywood, M. I. (2007). A hierarchical som-based intrusion detection system. *Engineering Applications of Artificial Intelligence*, **20**, 439–451.
- Kohonen, T. (1991). *Self-Organizing Maps: Optimization Approaches*. Artificial Neural Networks. North-Holland, 981–990.
- Lampinen, J. and Oja, E. (1992). Clustering properties of hierarchical self-organizing maps. *Journal of Mathematical Imaging and Vision*. Springer, **2(2–3)**, 261–272.
- Lampinen, J. and Kostiaainen, T. (1999). Overtraining and model selection with the self-organizing map. In *International Joint Conference on Neural Networks, IJCNN '99*. **3**, 1911–1915.
- Lampinen, J. and Kostiaainen, T. (2000). Self-organizing maps in data analysis-notes on overfitting and overinterpretation. In *ESANN*, Citeseer, 239–244.

- Lawrence, S., Giles, C. L. and Tsoi, A. C. (1997). Lessons in neural network training: Overfitting may be harder than expected. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence, AAAI'97*, AAAI Press, 540–545.
- Liu, Y., Zhong, Zh., Wang, G. and Hu, D. (2015). Cyclostationary Detection Based Spectrum Sensing for Cognitive Radio Networks. *Journal of Communications*, **10(1)**, 74–79.
- Moreira, M. and Fiesler, E. (1995). Neural networks with adaptive learning rate and momentum terms.
- Pateritsas, C., Pertselakis, M. and Stafylopatis, A. (2004). A som-based classifier with enhanced structure learning. In *IEEE International Conference on Systems, Man and Cybernetics, 2004* **5**, 4832–4837.
- Stasionis, L. and Serackis, A. (2015). A new self-organizing map topology for real-time spectrum sensing and fast convergence. In *IEEE International Conference on Computer as a Tool (EUROCON 2015)*, 1-4.
- Solodov, M. and Svaiter, B. (2000). A comparison of rates of convergence of two inexact proximal point algorithms. In G. Pillo, & F. Giannessi (Eds.), *Nonlinear Optimization and Related Topics*, Springer US **36** of *Applied Optimization*, 415–427.
- Weijters, T., Van Den Herik, H. J., Van Den Bosch, A. and Postma, E. O. (1997). Avoiding overfitting with bp-som. In *IJCAI*, Citeseer, 1140–1145.
- Yin, H. (2008). The self-organizing maps: Background, theories, extensions and applications. In J. Fulcher, and L. Jain (Eds.), *Computational Intelligence: A Compendium*, Springer Berlin Heidelberg **115** of *Studies in Computational Intelligence*, 715–762.
- Yucek, T. and Arslan, H. (2009). A survey of spectrum sensing algorithms for cognitive radio applications. *Communications Surveys & Tutorials, IEEE*, **11**, 116–130.

Received September 22, 2015 , revised December 7, 2015, accepted December 21, 2015