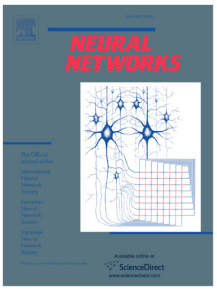
Accepted Manuscript



Self-organizing maps with dynamic learning for signal reconstruction

Jeongho Cho, Antonio R.C. Paiva, Sung-Phil Kim, Justin C. Sanchez, Jose C. Principe

PII: S0893-6080(06)00278-4
DOI: 10.1016/j.neunet.2006.12.002
Reference: NN 2247

To appear in: *Neural Networks*

Received date: 26 July 2005
Revised date: 7 December 2006
Accepted date: 7 December 2006

Please cite this article as: Cho, J., Paiva, A. R. C., Kim, S. -P., Sanchez, J. C., & Principe, J. C. Self-organizing maps with dynamic learning for signal reconstruction. *Neural Networks* (2006), doi:10.1016/j.neunet.2006.12.002

SELF-ORGANIZING MAPS WITH DYNAMIC LEARNING FOR SIGNAL RECONSTRUCTION

Jeongho Cho, Antonio R.C. Paiva, Sung-Phil Kim, Justin C. Sanchez and Jose C. Principe*

Computational NeuroEngineering Lab.,
University of Florida, Gainesville, FL 32611
{jeongho, arpaiva, phil, justin, principe}@cnel.ufl.edu

Abstract

Wireless Brain Machine Interface (BMI) communication protocols are faced with the challenge of transmitting the activity of hundreds of neurons which requires large bandwidth. Previously a data compression scheme for neural activity was introduced based on Self Organizing Maps (SOM). In this paper we propose a dynamic learning rule for improved training of the SOM on signals with sparse events which allows for more representative prototype vectors to be found, and consequently better signal reconstruction. This work was developed with BMI applications in mind and therefore our examples are geared towards this type of signals. The simulation results show that the proposed strategy outperforms conventional vector quantization methods for spike reconstruction.

Keywords: Spike Reconstruction, Self-Organizing Map, Brain-Machine Interface, Vector Quantization

Submitted to Neural Networks as a Regular Paper.

Please address correspondence to:
Jose C. Principe, Ph.D.
EB451 Engineering Bldg.
PO Box 116130
Gainesville, FL 32611
Phone: +1-352-273-5409
Fax:1-352-273-5411
Email: principe@cnel.ufl.edu

1. INTRODUCTION

Brain-Machine Interfaces (BMIs) have recently emerged as a technology to create real-time interfaces between living neuronal tissue and rehabilitative devices (prosthetics, computers) for restoring motor and sensory function in disabled individuals. Recent experimental paradigms performed in rodents, primates, and human patients have shown the feasibility of microelectrode technology, signal processing decoding algorithms, as well as functional organism/machine interaction [1]. The control signals used as inputs for external devices are extracted from hundreds of electrodes chronically implanted in the motor, premotor, and parietal cortices and useful trajectory reconstructions have been achieved through the use of local field potentials, spike waveform analysis (sorting), and firing rates [2].

Recently the appeal for developing wireless neuronal data transmission protocols has gained considerable attention. In the clinical setting, wireless technologies will enable increased mobility. Moreover, for subcutaneous BMIs, wireless will help reduce the risk of infection from wires traveling through the skin. For animal studies, wireless technology would free cumbersome wired behavioral paradigms where the experimenter must overcome issues of entanglement, torque applied to the prosthetic, and chewing of the cable.

To recover the maximal amount of electrophysiological information from the neuronal tissue, the bandwidth required for transmitting raw digitized potentials can be very large which results from the high channel count and the high bit rate in each channel. For instance, with 32 channels sampled at 20 kHz with 16 bps, the total bandwidth required in the link exceeds 9 Mbps, which results in high power consumption. In order to be able to overcome this hurdle, it is necessary to devise an efficient compression method. One possible approach would be to perform the spike detection onsite and then transmit only the wave shape or time that the spike occurred. Each of these can reduce the bandwidth considerably. Implementing a binning procedure would reduce the bandwidth significantly (i.e., counting of the number of spikes in a channel during a

given time interval) and would seem like a viable solution since most current neural decoding algorithms use only the firing rate of neurons [11][12]. Each of these previously proposed techniques is faced with the nontrivial tasks of spike detection and even sorting before transmission. Moreover, transmitting only spike times or binned data does not allow for any post processing at a later time since information about the original waveform has been discarded. Doing spike detection onsite and transmit some information on the spike allows some degree of spike sorting but usually the compression ratios are either too small, if all the samples of the spike are transmitted [4], or the reconstruction is too crude [3][5].

The large bandwidth gap between raw potentials and spikes has motivated the search for compression methods that act upon the full neuronal recording. The approach presented by Paiva et al. [13] seeks to transmit as much information about the waveforms of the neuron potentials as possible, under the constraints on low bandwidth and low complexity. So, rather than using just spike instants or spike counts, their approach seems more plausible for the reconstruction of the signals. This also means that the responsibility of spike detection and sorting is now transferred away from the neuronal environment and to some external processing unit where power and size constraints may not be so strict. This allows for more advanced spike detection and sorting signal processing algorithms to be utilized. Their strategy and mechanism for compression is to transmit through the wireless link only the index determined by the SOM, which encodes the raw digitized potentions.

The SOM has the characteristic of being a local framework liable to limit the interference phenomenon and to preserve the topology of the data using neighborhood links between processing elements (PEs). Neighboring PEs in the network compete with each other by means of mutual lateral interactions, and develop adaptively into specific detectors of different signal patterns [8]. The training algorithm is simple (although computationally intensive), robust to missing values, and it is easy to visualize the map. These properties make the SOM a prominent tool in data mining [14][16]. Compared to other Vector Quantization (VQ) [6][15] methods

which disregard neighborhood relations, e.g. k-means [6][18], LVQ [17], Fuzzy c-means [7], etc, the SOM may introduce performance improvements from the utilization of the topology [8].

For any network topology, training is a significant issue while the traditional SOM training techniques may not be the best for spike waveform reconstruction. It was shown in Paiva et al. [13] that due to the sparseness of spikes in the recordings, the spike reconstruction is accomplished with a relatively small number of PEs, and thus the peaks of many spikes are not well captured. This poses a problem especially for spike sorting [4] which uses the amplitude as a feature. Therefore, we are proposing a new SOM training algorithm with dynamic learning (SOM-DL) that is designed to increase the number of PEs to represent spikes and decrease the quantization error in their reconstruction.

Simulations with synthetic and real spike trains using the proposed strategy for spikes reconstruction are presented to demonstrate the versatility of the algorithm. Results show that the proposed SOM-DL methodology is a promising alternative for spikes reconstruction when compared with the regular SOM and other vector quantization methods. Moreover, SOM-DL offers a better compression ratio for neuron potential reconstruction when compared with the regular SOM.

2. Description of Wireless Data Transmission System for BMIs

Figure 1 presents a block diagram of the main elements of the communication process considered in this paper, and specifically showing the base elements of the encoder and decoder. Compression is achieved through a two step process: firstly, through quantization of the input vector by the SOM and, secondly, through entropy coding of the index determined by the SOM. The wireless data transfer procedure begins by grouping M samples from the input such as to form a vector, in which each sample is a component of this vector. This can be thought as a non-overlapping sliding window over the input. This vector is then applied as input to the SOM,

resulting in the index of the firing PE for this vector. Then, the index is entropy coded and transmitted through the communication channel. Conversely, at the receiver, the index is first entropy decoded and used as the reference to a lookup table (prototype vectors). It is important to note that the SOM, acting as a vector quantizer, insures through its competitive strategy that the winning PE is the one which is nearest to the input vector. The corresponding prototype vector of the selected PE is then used as a good approximation of the input vector at the receiver. Regarding entropy coding, any method can be used, such as, arithmetic coding or Huffman coding. One of the advantages of this separation between steps is that each step can be improved and/or adjusted according to specific requirements with coupling only through the distribuition of the indices. However, typically due to the specific characteristics of each signal it is not possible to estimate in advance the probability density function (PDF) of the indices, thus an adaptive method is needed. This, nevertheless, poses little restrictions since most entropy coding methods have an adaptive version [9][10].

3. SOM WITH DYNAMIC LEARNING

The principal goal of the SOM is to map an incoming signal pattern of arbitrary dimension into a one or two-dimensional discrete map, and to perform this mapping adaptively in a topologically ordered fashion [8]. Suppose the SOM consists of a set of *N* PEs. Each PE in the lattice is fully connected to all the PEs in the input layer. In other words, this network represents a feedforward structure with a single computational layer consisting of PEs arranged in rows and columns.

Let the input space be denoted by $\vec{x}_k \in \Re^M$. The synaptic weight vector, $\vec{w}_i, i \in \{1, \cdots, N\}$, of each PE i in the network has the same dimension of M, as the input space. To find the best match of the input vector $\vec{x}_k$ with the weight vectors $\vec{w}_i$, the Euclidean distance between $\vec{x}_k$ and each $\vec{w}_i$ is computed. Then the index of the PE with the smallest distance is selected as

$$i^o = \underset{1 \le i \le N}{argmin} \left\| \vec{x}_k - \vec{w}_i \right\| \tag{1}$$

which summarizes the essence of the competition process among the PEs. The particular PE that satisfies this condition is called the winning PE, i^o, for the input vector $\vec{x}_k$. The winning PE locates the center of a topological neighborhood of cooperating PEs. The weight vector $\vec{w}_i$ of each PE is then updated as

$$\vec{w}_{i,k+1} = \vec{w}_{i,k} + \eta_k \Lambda_{i,k} (\vec{x}_k - \vec{w}_{i,k}) \tag{2}$$

where $\eta_k \in [0,\ 1]$ is the learning rate. $\Lambda_{i^o,i}$ is the topological neighborhood function centered on the winning PE i^o, and is typically defined as $\Lambda_{i^o,i,k} = exp\left(- \left\| r_i - r_{i^o} \right\|^2 \Big/ 2\sigma_k^2 \right)$, where $\left\| r_i - r_{i^o} \right\|$ represents the Euclidean distance in the output lattice between the i^{th} PE and the winning PE. σ_k is the effective width of the topological neighborhood. Notice that both the learning rate (η_k) and the neighborhood width (σ_k) are time dependent and are normally annealed (from large to small) to provide the best performance with the least training time.

The adaptation rule (2), however, may not work appropriately for neural firing activity signals since the SOM puts more PEs in the region of the space where the density of samples is the highest, instead of minimizing the reconstruction error from spikes as is required by our application. Specifically, spikes are represented by few PEs since they are fast events (i.e. few samples represent spikes). Even if we isolate the spike waveforms for training, the percentage of samples corresponding to the actual spike peaks still remains small so that the input vectors tend to be close to the origin of state space. Since we are interested in using the smallest number of PEs for higher compression in the configuration of the wireless communication system, very few PEs will be assigned to accurately represent the spike waveshape. Thus, our objective is to move PEs from the low amplitude region of state space to the one corresponding to the spikes.

Although this may result in larger errors in the background activity the representation of our most important signal feature (spikes) is ensured.

Hence, we modify the adaptation rule (2) as follows;

$$\vec{w}_{i,k+1} = \vec{w}_{i,k} + \mu\Lambda_{i,k} sign(\vec{x}_k - \vec{w}_{i,k})(\vec{x}_k - \vec{w}_{i,k})^2 \quad (3)$$

and name it as Self-Organizing Map with Dynamic Learning (SOM-DL). A simple example illustrating the proposed SOM-DL algorithm comparing with the regular SOM is shown in Figure 2. In the regular SOM, the weight vector is updated equally with a constant learning rate, η, during one *epoch*. On the other hand, the proposed training rule thru the square modifies the learning rate *for the components* of the weight vector proportional to the distance between itself and the sample. The term *sign*(·) operates component-wise and preserves the movement of the PEs towards the current input vector. Note that the topological neighborhood will still be preserved by $\Lambda_{i^o,i}$, so that a winning PE tends to excite the PEs in its immediate neighborhood more than those farther away from it.

In (3), μ is the learning rate to control the speed of convergence. If μ is close to 1, then the training is very fast and more PEs move to the region where the spikes are mostly distributed but the network may never stabilize. In contrast, by choosing μ small, the map is formed even slower than the regular SOM, and spikes become difficult to capture. Thus a suitable choice of μ makes the network training faster as well as generates better precision in the region of interest. It was experimentally verified that a value of μ between 0.05 and 0.5 balances between fast convergence and small quantization error for the spikes.

4. CASE STUDIES

In this section, we provide three examples demonstrative of the capabilities and of a possible SOM-DL algorithm application. In the first two examples, synthetic data are used to

benchmark the training algorithm under a controlled environment. The first example shows the better assignment of the PEs through the vector space. The second and third examples both employ neuronal activity waveforms albeit in the third example real waveforms were used. These last two examples intend also to support the efficacy of the SOM-DL algorithm to real world problem such as the reconstruction of neural spike signals, which initially motivated this development.

Example 1. Reconstruction of a sparse sequence of pulses

For the demonstration of the proposed SOM-DL algorithm, we generated a waveform comprised of a sparse sequence of pulses. Our goal with this example is to show the capability of the proposed algorithm to improve on the waveform reconstruction of sparse events. This was achieved through a better distribuition of the PEs in the vector space, as claimed before.

The generated waveform assumes a sampling frequency of 20 kHz and was formed with pulses of amplitudes 0.8 and 1, distributed through time according to a Poisson process with firing rates 70 and 30 pulses/s, respectively. All pulses were 0.75 ms (15 samples) long. The amplitude of each pulse was disturbed with a Gaussian distributed random variable of mean 1 and standard deviation 0.025. The entire waveform was further contaminated with Gaussian distributed zero mean white noise with standard deviation 0.05. A total of 2 seconds of this waveform were generated, with the first 0.5 seconds being used for training and the remaining for testing.

The distribuition of the SOM input (x_1 , x_2) obtained from the embedding of the waveform in a two dimensional space is shown in Figure 3. There are three main clusters, corresponding to the “no pulse” activity and each of the two amplitude pulses, and two small clusters along the axis but away from the origin correspond to the transitions at the beginning and

end of a pulse. Also, in the same figure the prototype vectors of a 5×5 square lattice SOM trained with the general adaptation rule, with learning rate at the *k*-th epoch $\eta_k = 0.1/(1+0.006k)$, and with the proposed dynamic learning rule with $\mu = 0.1$, over 4000 epochs are shown in (a) and (b), respectively. The width of the topological neighborhood function at the *k*-th epoch was $\sigma_k = b/(1+0.003k)$, with *b* equal to 5 and 10/3 for the regular SOM and SOM-DL, respectively. Analysing the distribuition of the prototype vectors over the space, we verify that most of the data vectors are concentrated around the origin and, consequently, in both cases most PEs are assigned to this region. It is noteworthy however that the SOM-DL training algorithm was capable of forcing at least one PE per cluster, unlike the regular SOM, allowing for the two pulse amplitudes to be distingushed. Also, PEs were assigned to the clusters corresponding to the signal transitions. We also evaluated topographic quality based on the topographic error [19][20] measured by Kohonen's method[1]; there is a distortion in the map if the winning PE that has the shortest Euclidean distance from the input is not an adjacent PE of the second winning PE. The topographic distortion produced from the regular SOM and the SOM-DL was 20.1% and 33.0%, respectively. This result is expected since both SOMs have the same number of PEs and the conventional SOM minimizes the mean distortion, which due to the sparseness of the pulses means that it is more productive for the cost to represent the low amplitude background well than to compensate for large errors in the infrequent pulses. SOM-DL on the other hand tries to minimize larger errors even if they occur infrequently producing a more constant error throughout the dynamic range of amplitudes (similar to the L1 norm). Therefore, the SOM-DL still preserves the data topology nearly as well as the regular SOM does.

[1] The topographic error is defined by $T\_Error = 1/M\left(\sum_{i=0}^{M}\{1, if \left\| r_{i^o} - r_{i^{o+}} \right\| > 1 \; and \; 0, otherwise\right)$ where r_{i^o} and $r_{i^{o+}}$ are the positions of the winning PE and the second winning PE for input *i* on the SOM, respectively.

The learning curves with respect to the largest quantization error ($\max_k \left\{ \min_i \arg \left\| \vec{x}_k - \vec{w}_{k,i} \right\| \right\}$, $i \in \{1, \cdots, N\}$) for both the regular SOM and the SOM-DL are shown in Figure 4. These curves reflect the closeness of the winning PE to the input samples, especially the sample whose magnitude is relatively larger than others of the input vector, during the training process and it becomes approximately constant at the end of training. As seen in the figure, the SOM-DL demonstrates faster convergence as well as lower maximum quantization error than the regular SOM.

The reconstructed waveforms in a test set by the SOM-DL and the regular SOM are presented in Figure 5. As seen in the figure, the better distribuition of the PEs through the space by the SOM-DL leads to a more faithful reconstruction of the waveform features, namely, the fact that it is possible to distinguish between the two pulse amplitudes. This is clearly expressed in the reconstruction SNR (15.8 dB and 20.6 dB for the regular SOM and SOM-DL, respectively). This occurs because the SOM puts resources (PEs) in the wrong place when features of sparely occurring waveforms need to be well represented.

Example 2. Reconstruction of synthetic neuronal activity

As mentioned at the beginning of this paper our main motivation was to better use small SOMs for quantization of neural activity. In this example, the reconstruction of synthetic neuronal activity is tested while working in a controlled environment where the characteristics of data are well under control.

The waveforms were assumed to be sampled at 20 kHz, and to contain action potentials (i.e. spikes) from two neurons differing in both peak amplitude and width of the spike as shown in Figure 6. Both neurons fired according to a homogeneous Poisson process with firing rates, 10 spikes/s (squares) and 20 spikes/s (circles). Also, to introduce some variability in the recorded template each time a neuron fired the template was scaled by a Gaussian distributed random

number with mean 1 and standard deviation 0.01. Finally, the waveform was contaminated with zero mean white noise of standard deviation 0.05. Unlike the previous example in which the training dataset was obtained directly from a segment of the waveform, here we are interested mostly in the reconstruction of spikes. For that reason, a separate training dataset was constructed with 100 spikes as if they had been segmented from a real waveform after spike detection, by removing all noise background between spikes. The ratio of spikes from each neuron was approximately the same as in the testing waveform. For testing, five seconds of synthetic neuronal activity were generated.

Figure 7 shows the distribuition of the SOM input after embedding of the waveform in a two dimensional space. From the figure the main cluster of points around the origin is noticeable as a consequence of the sparseness of the spikes and the overwhelming percentage of low voltage background activity, and a number of smaller, less dense, clusters scattered throughout the space. Unlike real waveforms were the points in state space corresponding to spikes are spread around a trajectory in rather uniform way, in this example these points are grouped around small clusters due to the repeatability of the generated spike train. The same figure shows the prototype vectors of a 5×5 square lattice SOM (Figure 7.(a)) trained with a learning rate at the *k*-th epoch $\eta_k = 0.1/(1+0.006k)$, and the SOM-DL (Figure 7.(b)) trained with $\mu = 0.1$, over 4000 epochs. The width of the topological neighborhood function was chosen with the same parameters an in the previous example. Analysing the distribuition of the prototype vectors over the space, we verify that for both training methods the proportion of PEs for spike reconstruction and "no activity" is comparable. This is a consequence of the construction of the training dataset which emphasizes the reconstruction of spike waveforms. However, it is noteworthy that even in the case of creating a training set only with the waveforms of interest, a more uniform distribution of prototype vectors is achieved with the SOM-DL. Yet, the topographic error produced was comparable (54.7% and 67.1% by the regular SOM and the SOM-DL, respectively). The increase

in these values is also because of the separate training dataset which bias the prototype vector distribuition from a pdf matching standpoint, but with higher spike reconstruction accuracy.

The learning curves with respect to the maximum quantization error for both methods are shown in Figure 8. As in the previous example, the SOM-DL demonstrates faster convergence as well as final lower maximum quantization error than the regular SOM.

The reconstructed waveforms in a test set by the SOM-DL and the regular SOM are presented in Figure 9. As highlighted in the figure, the spike waveform reconstructed with the SOM-DL algorithm is more accurate than with the regular SOM. For the reconstruction of spike waveform segments, the SNR is 14.6 dB and 16.8 dB for the regular SOM and SOM-DL, respectively. However, the overall SNR on the entire test segment is 9.77 dB and 8.60 dB for the regular SOM and SOM-DL, respectively, which highlights the "waste" of PEs to represent the featureless background activity.

Example 3. Reconstruction of (real) neuronal activity

Multielectrode array recordings were collected from male Sprague-Dawley rats performing a go-no go lever pressing task. Array configurations of $2\times8\times50\mu$m tungsten electrodes were chronically implanted in the forelimb region of M1 (+1.0mm anterior, 2.5mm lateral of bregma). Neuronal activity was collected with a Tucker-Davis recording rig with sampling frequency of 24,414.1Hz and digitized to 16 bits of resolution. Before being stored to disk the neuronal potentials were bandpass filtered between 0.3 and 7 kHz. From these recordings we considered only one channel in which the spike activity of neural population was detected under human supervision.

The data set for training the SOM was formed as a number of spikes from one channel and 2% (relative to the samples of spikes) of background activity samples which are originated from the channel. We used 8,000 spikes and 4,320 samples of background, totalizing 220,320

data samples assuming the average number of samples for a spike as 27. The reason of using the concatenated signal (spikes only) instead of the original signal is to put more emphasis on the spikes in the formation of the SOM map. Due to the density matching property of the SOM, only very few number of PEs would be assigned for the approximation of spikes if trained with the original signal. On the other hand, the presence of a small percentage of background activity samples makes the SOM "noise aware" and prevents PEs dedicated to approximate spikes from firing through background areas which would cause misinterpretation, at the decoder, on the content of the input waveform.

Both the regular SOM and the SOM-DL were trained in a 10 dimensional space with $\vec{x}_k = [x_k, \cdots, x_{k-9}]^T$ (i.e., $M = 10$) over 220,320 samples with the time decaying parameter $\sigma_k = (\sqrt{N}/1.5)/(1+0.004k)$ in (2), where N is the number of PEs. Other parameters chosen were $\eta_k = 0.1/(1+0.004k)$ in (2) and $\mu = 0.5$ in (3). The PEs corresponding to the spike data which generates relatively large error were updated until the error reaches a certain level since the SOM-DL uses a constant learning rate as well as error-power-based adaptation. Once again, Figure 10 shows the learning curves in terms of the largest quantization error and the advantage of the proposed SOM-DL scheme by showing faster convergence as well as much lower maximum quantization error than the regular SOM.

The topographic distortions measured from the regular SOM and the SOM-DL were, respectively, 11.46% and 11.07% showing, once again, that the SOM-DL is able to preserve the topology as the SOM does.

After training square SOMs (for both the regular SOM and the SOM-DL) with various sizes, the networks were tested in a new sequence of 400,000 samples. The reconstruction results are listed in Table 1 as a function of a threshold parameter γ on the samples magnitude, i.e. only for signal samples with magnitude greater than the threshold are considered in the error calculation. As expected from the simulations, we can see that the regular SOM produced a better

overall reconstruction result than the SOM-DL when all the samples are considered ($\gamma = 0$). However, this result does not hold when only large errors are considered. The second and third row ($\gamma > 0.2$) show that the SOM-DL has less power in the largest error percentiles, which means that a combination of less samples are fit with large errors, or the errors are small. Since the large errors tend to occur in the large waveform amplitudes that correspond to spikes, this means that spike representation improves with the SOM-DL. Moreover, an increase in the SOM size does not seem to improve the power in high errors (rows 2 and 3) in the conventional SOM training, while for the SOM-DL the power in the high error percentiles decreases with the SOM size.

Figure 11 shows the result of spike train reconstruction with the regular SOM and the SOM-DL. It is obvious from the figure that the SOM-DL represents the morphology of the spikes much better. In addition, we compared our proposed SOM-DL scheme with other VQ methods (K-means, Fuzzy C-means) in Figure 12. As can be seen, all the methods follow the trend of the SOM: they are slightly better than the SOM-DL for $\gamma = 0$ (all samples considered), but the SOM-DL startes to show its benefit as the threshold is increased. Additionally the difference between the SOM-DL and other VQ techniques is statistically significant with $p < 0.01$ (t-test). As a result, we can achieve better spike reconstruction using the SOM-DL rather than using other VQ approaches.

A different way of verifying the extent to which each algorithm preserves the spike shape is through spike sorting. We started by performed spike sorting on the original waveform using a widely used commercial software package, Spike2 (Cambridge Electronic Design, UK). These results were verified by an expert in spike sorting and used as the ground truth. *Using the spike templates obtained from sorting the original waveform*, we did spike sorting in the reconstructed waveforms (10×10 dimensional SOM lattice, 10-dimensional embedding). Then, the ratio of spikes correctly detected and, out of those, the ratio of spikes correctly sorted was measured. For

the SOM-DL 81.5% of the detected spikes were correct (found in the original sorting) whereas the regular SOM only achieved 74.4%. Moreover, for the spikes correctly detected sorting with the SOM-DL was 92.8% correct while for the regular SOM was 90.6%. In effect this means the SOM-DL preserves better the shape and introduces less artificial waveform features.

As initially stated our ultimate goal is to be able to compress as much as possible the spike train data, preserving the morphology of the spike waveforms for future analysis. An intermediate step is to improve the SOM training algorithm to allow better spike reconstruction, but this should not increase the overall data rates for transmission. The comparison of the histograms created by the regular SOM and the SOM-DL in Figure 13 confirm that the difference in performance is associated with a different assignment of PEs as a function of amplitude. In fact, the percentage of PEs representing low amplitude values (corresponding to background) decreased and moved to represent better high amplitude values typically associated with spikes. The net effect is to create an histogram of PEs that is less peaky than the SOM histogram (which follows the data amplitude histogram with a 2/3 rule as known [22]).

Given the significant changes in the prototype vectors it is expected the distribution of the PE firings will change. Therefore it is necessary to study the data compression of the new SOM-DL algorithm. As mentioned in Section 2, and depicted in Figure 1, the system involved the choice of an entropy coding algorithm. Naturally, the compression performance also depends on the properties of the entropy coding algorithm. Nevertheless, to maintain the generality of our results we simply computed the entropy of the indices as $h = -\sum_{i=1}^{N} p_i \log_2 p_i$, where p_i is the relative frequency of the index. Then, the entropy (i.e., the average number of bits per index) was used to calculate the compression ratios, $8M / h$,[2] assuming 8 bps. Table 2 presents the compression results. Considerably better compression ratios are achieved by the SOM-DL than the regular SOM as the number of PEs decrease. In Figure 14 the histogram of the frequency of

[2] The compression ratio is simply the ratio between the number of uncompressed bit needed to represent the data and the number of bits after compression for the same amount of data.

firing is shown. The histogram explains why these better compression results are obtained with the proposed SOM-DL by showing that the number of PEs assigned by the SOM-DL to cover the background region was decreased considerably when compared to the regular SOM. In other words, the SOM-DL algorithm assigned more PEs to the reconstruction of spikes, yielding better spike reconstruction. At the same time, the number of PEs for background activity was dimished and since these PEs fire more frequently their relative probability was increased. Consequently, the entropy decreased and thus the higher compression ratio was achieved.

5. CONCLUSIONS

In this paper, an improved spike reconstruction strategy has been proposed for wireless brain-machine interface topology. The architecture considered utilizes the SOM, for the vector quantization, followed by entropy encoding of the indices of the firing PEs. The regular SOM, however, revealed problems with the accuracy of the reconstruction of spike morphology since they occur very sparsely in the data, and they contain most of the information for this application. The conventional SOM emphasizes the reconstruction accuracy of the most probably samples in the data, so a relatively small number of PEs are assigned to represent spikes due to their sparseness. Thus, the regular SOM adaptation rule has been modified to include a dynamic learning rate factor that emphasizes larger errors in order to reduce the quantization error associated with the reconstruction of portions of the signal that display large and infrequent errors. This issue is of especially important for post processing which depends directly on the accuracy of spike reconstruction (e.g. spike detection and sorting). The effectiveness of the proposed approach was quantified through synthetic and real data experiments for neuronal activity reconstruction. Example 1 shows that very poor results are obtained when the data is taken directly from segments of the signal source. As initially proposed by Paiva et al. [13], and employed in examples 2 and 3, an acceptable result can be achieved with the regular SOM by creating a training dataset that emphasizes the sparse elements in the waveform that we are most

interested in. However, even in this case a statistical significant improvement of performance is obtained with the proposed SOM-DL training because the method is more sensitive to the error power in the higher percentiles. Therefore, an important advantage of using the proposed learning rule is the weaker dependence on the selection of the training dataset when the features of interest are sparse.

A comparison with other VQ alternatives (e.g. k-means, Fuzzy c-means) showed clear advantages of the proposed SOM-DL algorithm in terms of spike reconstruction performance. And, most importantly, the improvement is visible in terms of spike detection and sorting with respect to the regular SOM. An additional advantage of the proposed scheme is that it greatly increases the compression ratio when compared with the method based on the regular SOM. It is also important to highlight that the advantages described comes at nearly no additional cost.

In spite of our successful results in terms of spike detection and sorting further work is needed in improving reconstruction accuracy. In any case, the high compression ratios already achieved proved the framework chosen to be highly appropriate and that reconstruction accuracy is one of the issues holding back the method. High training computational complexity and application of the vector quantization (due to the SOM search) are also very significant problems since low complexity / low power implementations are desired.

Acknowledgements: This work was partially supported by DARPA under grant ONR-450595112. Antonio R. C. Paiva was supported by Fundação para a Ciência e a Tecnologia under grant SFRH/BD/18217/2004.

REFERENCES

[1] J. Wessberg, C.R. Stambaugh, J.D. Kralik, P.D. Beck, M. Laubach, J.K. Chapin, J. Kim, S.J. Biggs, M.A. Srinivasan, and M.A. Nicolelis, "Real-time prediction of hand trajectory by ensembles of cortical neurons in primates," *Nature*, 408(16):361-365, 2000.

[2] D.M. Taylor, S.I.H. Tillery, and A.B. Schwartz, "Direct cortical control of 3d neuroprosthetic devices," *Science*, 296(5574):1829-1832, 2002.

[3] K.D. Wise, D.J. Anderson, J.F. Hetke, D.R. Kipke, and K. Najafi, "Wireless implantable microsystems: high-density electronic interfaces to the nervous system," *Proceedings of the IEEE*, 92(1):76-97, 2004.

[4] C.A. Bossetti, J.M. Carmena, M.A.L. Nicolelis, and P.D. Wolf, "Transmission latencies in a telemetry-linked brain-machine interface," *IEEE Trans. Biomedical Engineering*, 51(6):919-924, 2004.

[5] R. Olsson and K. Wise, "A three-dimensional neural recording microsystem with implantable data compression circuitry," in *Proc. IEEE Int. Solid-State Circuits Conf.*, pp.558-559, 2005.

[6] A. Gersho and R.M. Gray, *Vector Quantization and Signal Compression*. Kluwer Academic Press/Springer, Boston, MA, 1992.

[7] J.C. Bezdek, *Pattern Recongnition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, NY, 1981.

[8] T. Kohonen, *Self-Organizing Maps*, Springer-Verlag, 1995.

[9] K. Sayood, *Introduction to data compression*, Morgan Kaufmann, 1996.

[10] D. Salomon, *Data compression: the complete reference*, Springer-Verlag, 2000.

[11] M.S. Serruya, N.G. Hatsopoulos, L. Paninski, M.R. Fellows, and J.P. Donoghue, "Brain-machine interfaces: instant neural control of a movement signal," *Nature*, 416:141-142, 2002.

[12] S. Musallam, B.D. Corneil, B. Greger, H. Scherberger, and R.A. Andersen, "Cognitive control signals for neural prosthetics," *Science*, 305(5681): 256-262, 2004.

[13] A.R.C. Paiva, J.C. Principe, and J.C. Sanchez, "Compression of spike data using the Self-Organizing Map," in *Proc. IEEE EMBS Neural Engineering Conf.*, 2005.

[14] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen, "WEBSOM-Selforganizing maps of document collections," *Neurocomputing*, 21:101-117, 1998.

[15] R. M. Gray, "Vector quantization," *IEEE ASSP Mag.*, pp. 4-29, 1984.

[16] T. Kohonen, "Exploration of very large databases by self-organizing maps," in *Proc. Int. Conf. Neural Networks*, pp.PL1-PL6, 1997.

[17] T. Kohonen, *Self-Organization and Associative Memory*, Springer-Verlag, 1989.

[18] Y. Linde, A. Buzo, and R.M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Communication.*, 28:84-95, 1980.

[19] H. Bauer and K.R. Pawelzik, "Quantifying the neighborhood preservation of self-organizing feature maps," *IEEE Trans. Neural Networks*, 3(4):570-579, 1992.

[20] E. Erwin, K. Obermayer, and K. Schulten, "Self-organizing maps: Ordering, convergence properties and energy functions," *Biol. Cyb.*, 67(1):47-55, 1992.

[21] D.C. Montgomery and G.C. Runger, *Applied Statistics and Probability for Engineers*, John Wiley & Sons, New York, NY, 1994.

[22] Jens Christian Claussen, "Winner-relaxing Self-Organizing Maps," *Neural Computation*, 17:996-1009, 2005.

List of figures

List of tables

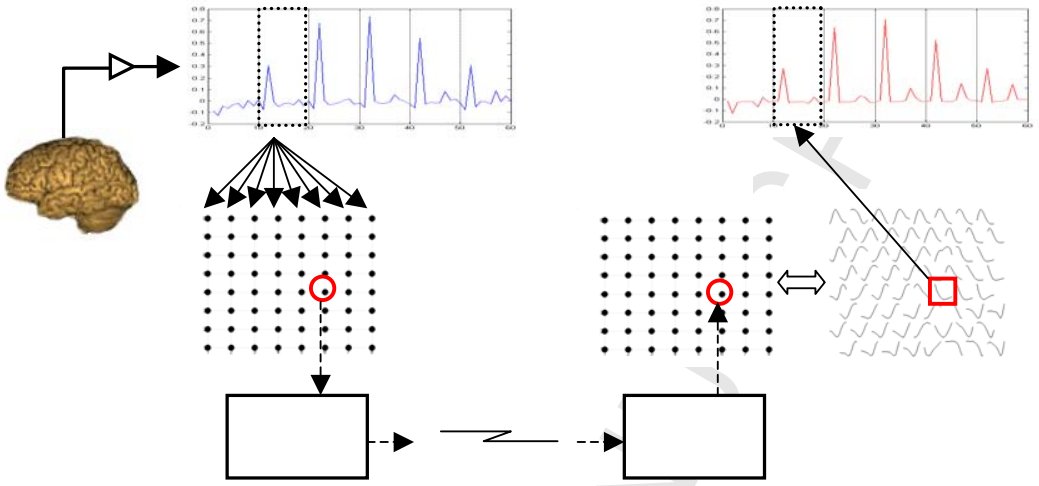
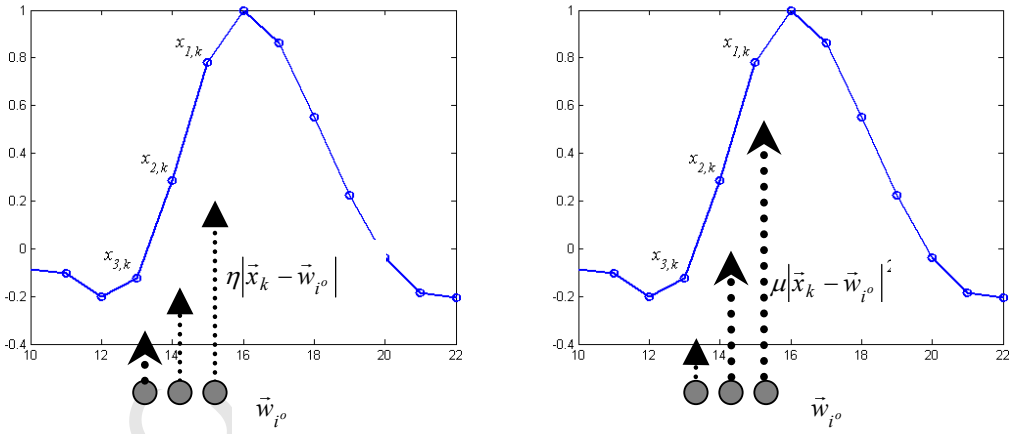
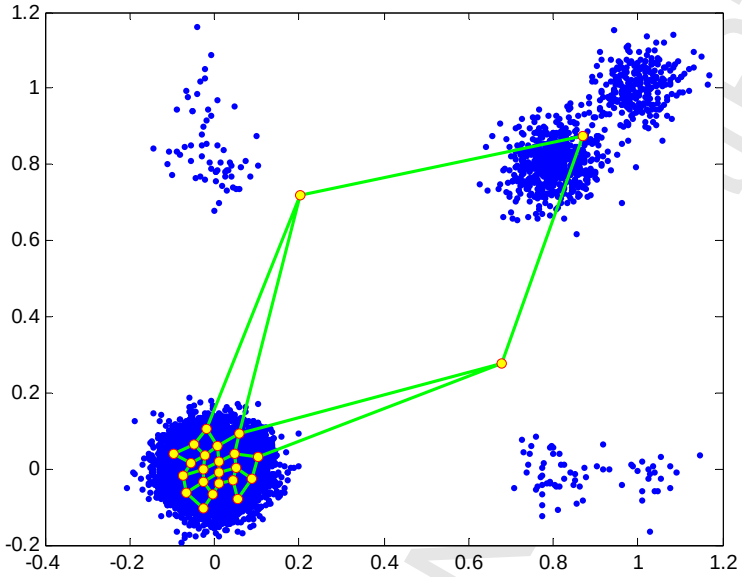


Figure 1. Schematic diagram of the wireless communication process for BMI.

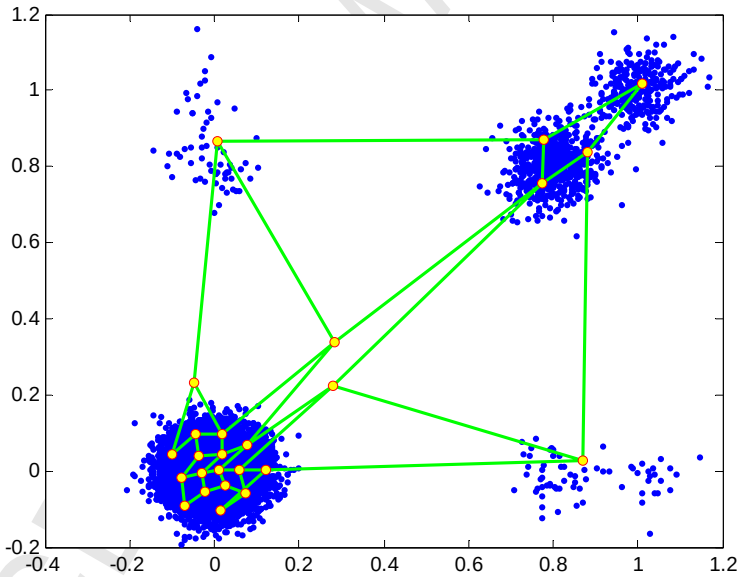


(a) (b)

Figure 2. An example to demonstrate the geometric meaning of (a) the regular SOM adaptation rule and (b) the proposed SOM-DL adaptation rule; bold dotted line represents more adaptation relative to the others.



(a)



(b)

Figure 3. Data distribution and the map formation of example 1 by (a) the regular SOM and (b) the SOM-DL.

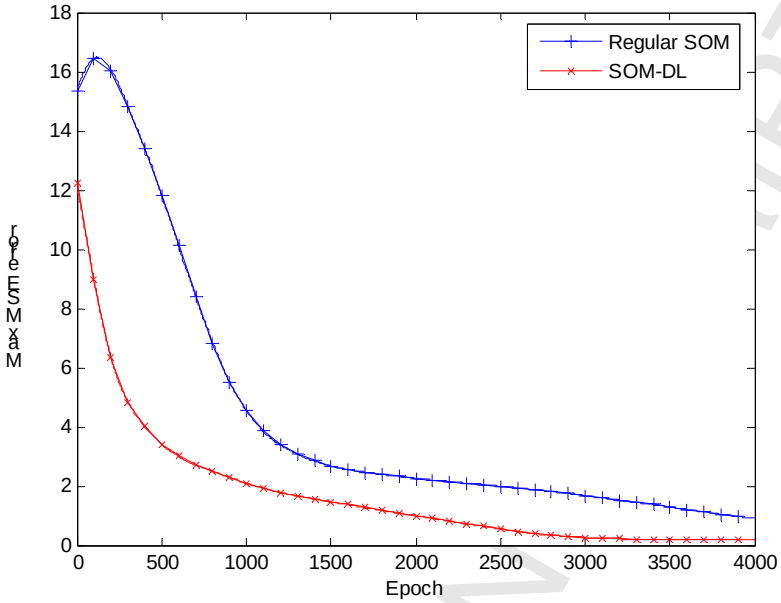


Figure 4. Learning curves of example 1 by the regular SOM (+) and the SOM-DL (×), with respect to the largest quantization error.

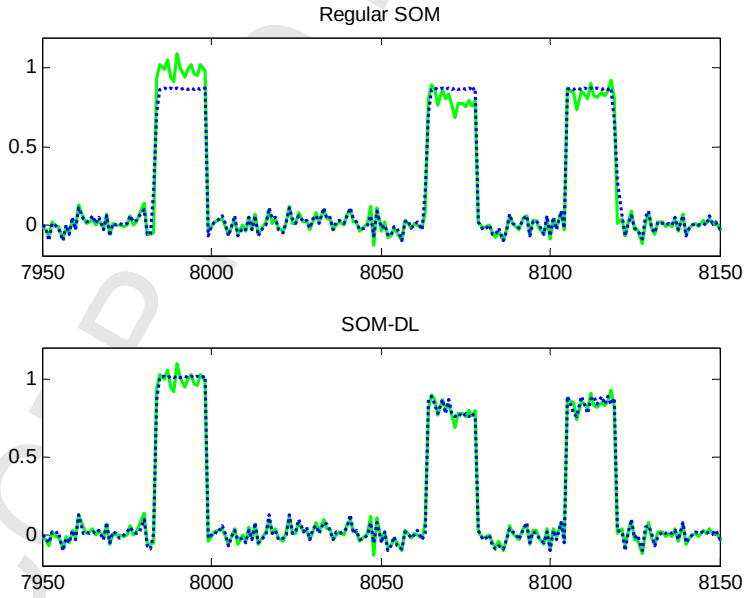


Figure 5. Segment of the original waveform (solid green) and reconstructed waveform (dotted blue) of example 1 by the regular SOM (top) and the SOM-DL (bottom).

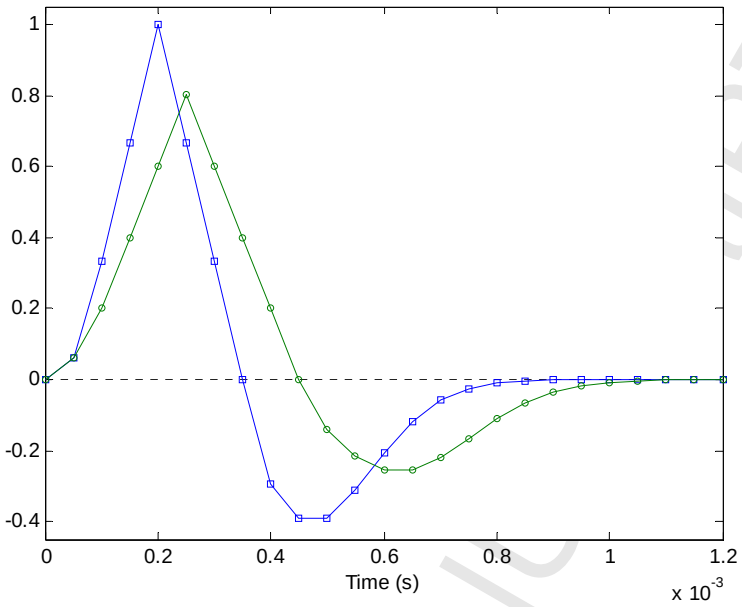
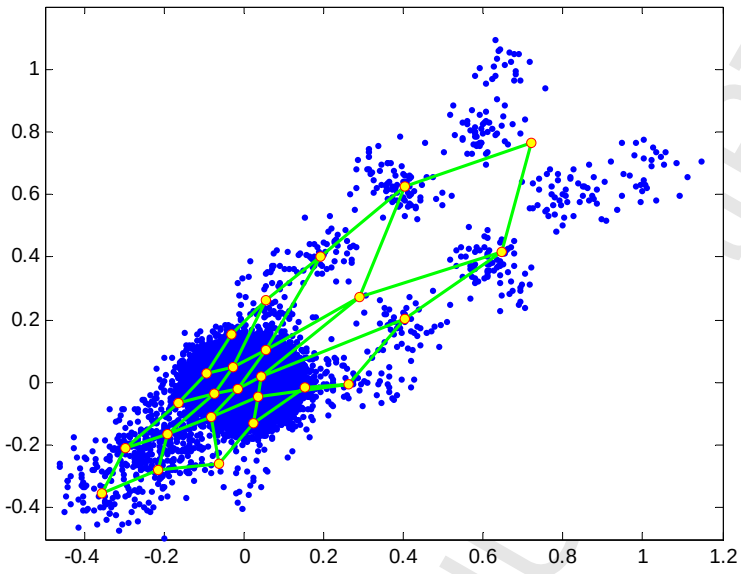
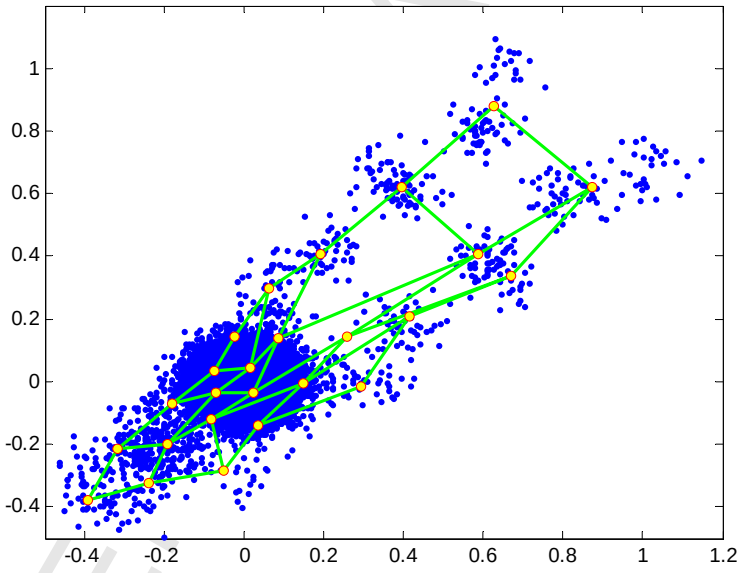


Figure 6. Spike templates used in the creation of both training and testing datasets for example 2.



(a)



(b)

Figure 7. Data distribution and the map formation of example 2 by (a) the regular SOM and (b) the SOM-DL.

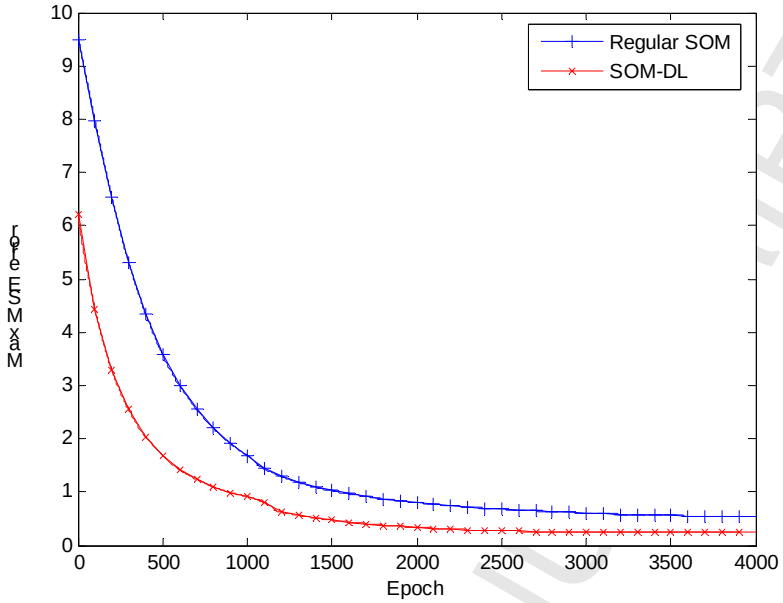


Figure 8. Learning curves of example 2 by the regular SOM (+) and the SOM-DL (×), with respect to the largest quantization error.

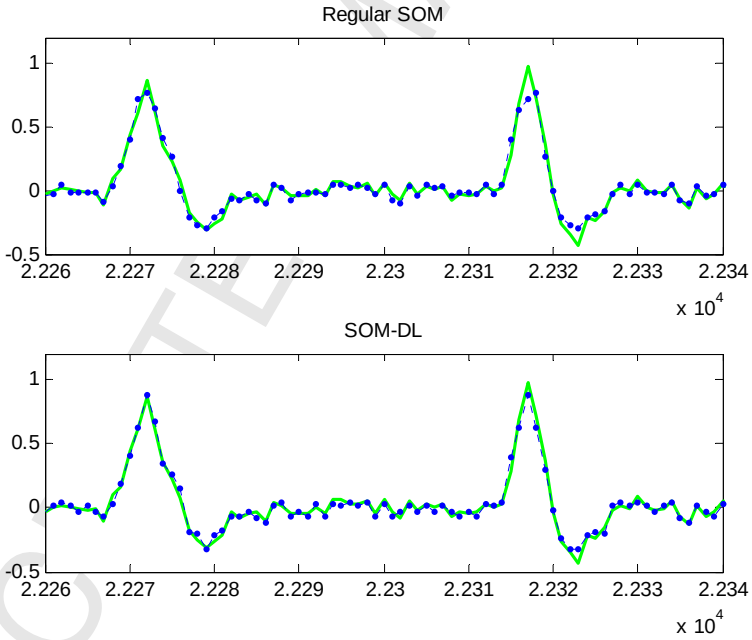


Figure 9. Segment of the original (solid green) and reconstructed synthetic neuronal activity (dotted blue) of example 2 by the regular SOM (top) and the SOM-DL (bottom).

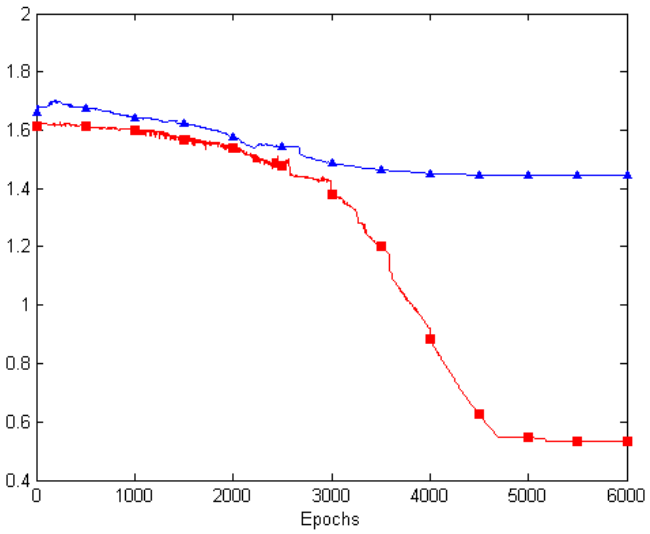
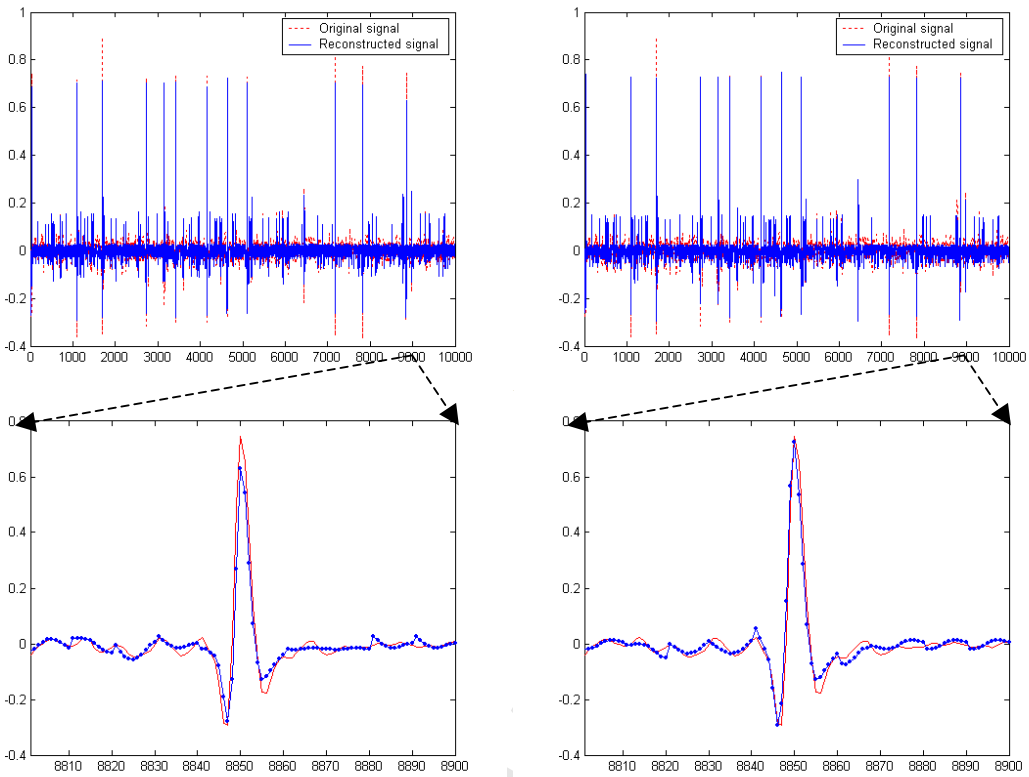


Figure 10. Learning curves of example 3 by the regular SOM (▲) and the SOM-DL (■).



(a) (b)

Figure 11. Original and reconstructed signal of example 3 using (a) the regular SOM and (b) the SOM-DL.

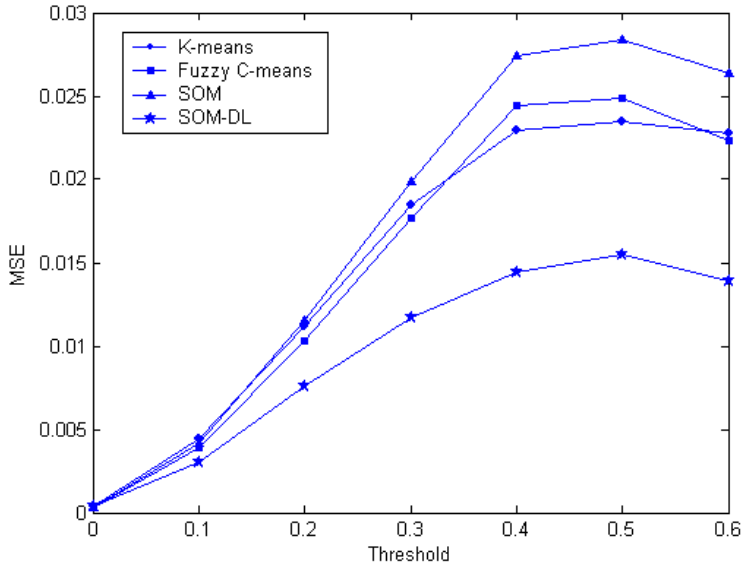
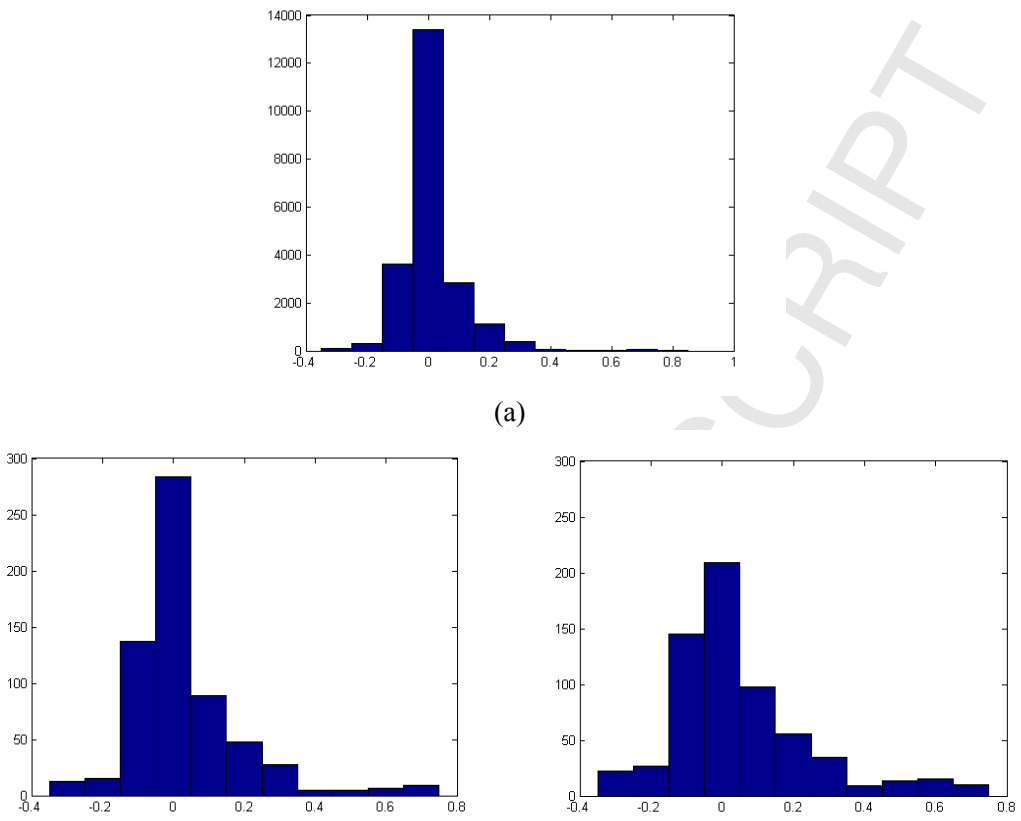


Figure 12. Comparison of reconstruction performance in example 3 by various VQ techniques with respect to the threshold.



(a)

(b) (c)

Figure 13. Histogram of (a) the original data and the prototype weights created by (b) the regular SOM and (c) the SOM-DL for example 3.

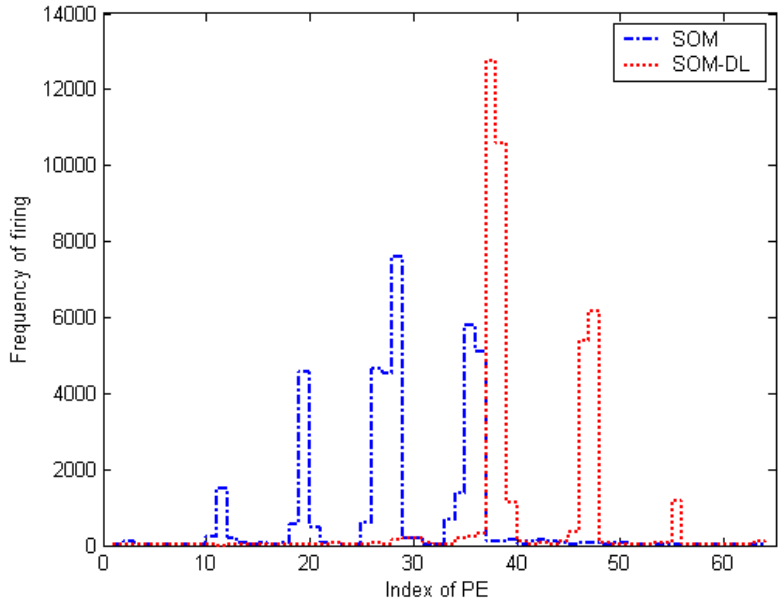


Figure 14 Histogram of frequency of firing using the regular SOM and the SOM-DL.

Table 1. Comparison of reconstruction performance by the regular SOM and the SOM-DL as a function of map size and the threshold on the magnitude of the signal samples.

Threshold	SOM Type	Dimension of the lattice (Number of PEs)			
		8×8 (64)	10×10 (100)	12×12 (144)	14×14 (196)
$\gamma = 0$	SOM	3.49e-4	2.91e-4	2.32e-4	2.12e-4
	SOM-DL	4.14e-4	3.14e-4	2.86e-4	2.33e-4
$\gamma = 0.2$	SOM	11.5e-3	10.2e-3	10.2e-3	10.3e-3
	SOM-DL	7.6e-3	6.2e-3	5.2e-3	4.9e-3
$\gamma = 0.4$	SOM	2.74e-2	2.44e-2	2.62e-2	2.71e-2
	SOM-DL	1.44e-2	1.32e-2	1.15e-2	1.14e-2

Table 2. Comparison of compression capability between the regular SOM and the SOM-DL as a function of the map size.

Compression Ratio		Reconstruction Methods	
		SOM	SOM-DL
Size of a map (Number of PEs)	64	32.1:1	42.2:1
	100	27.5:1	32.4:1
	144	22.1:1	30.1:1
	196	19.9:1	25.0:1