International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol. 2, No. 4, December, 2009

Principal temporal extensions of SOM: Overview

Mohamed Salah Salhi, Najet Arous, and Noureddine Ellouze

Laboratory of the Systems and Treatment of Signal (LSTS)
National school Engineers of Tunis, LP 37, the Belvédère, 1002 Tunis, Tunisia
Salhi_mohamedsalah@yahoo.fr
(Najet arous@yahoo.fr, N. Ellouze@enit.rnu.tn)

Abstract

The resolution of multy variable complex problems such as the multy speaker speech recognition and independently of context, requires the application of neural structures. One tool which proves to be powerful in the classification field, is the kohonen map called SOM. This map is characterized by the representation of static data. Thus, we ought to enrich the intelligibility and the performance of this model in order to reach what biology imposes by handling a kind of logical pile "memory" with the introduction of temporal context which realise feedbacks that integrate respectively the leaky integrators concept for the TKM. The recurrent leaky integrators idea for the recurrent SOM 'RSOM' in an improvement of the TKM. More recently, the principle of self reference for the recursive SOM. In other case, we are introducing the possibility to obtain hybridization with GA in an attempt to reach the natural evolution of the human thought as regards to recognition.

Keywords: *Kohonen map SOM, Temporal kohonen mapTKM, Recurrent SOM:RSOM, Recursive SOM, Growing RSOM:GRSOM, Temporal extensions of SOM, Hybridization of SOM with GA:GASOM.*

1. Introduction

Man always seeks to copy what nature does so well in various fields. For the resolution of multy variable complex problems he uses the brain. One of the models derived from the biology of the brain and which represents a very powerful tool in the classification of static data and thereafter in the speech or pattern recognition which was the kohonen map (1982;1995; 1997, and 2001), shortened in SOM for self organizing map. However, this map SOM remains limited to static data and does not treat the dynamic regularities (aspects) of a signal. To confines with this problematic, on the basis of the fact that all human activities are closely related to the time dimension, we tried to present in this overview, according to the order of sections, the most significant temporal extensions for this model knowing that TKM, the recurrent SOM, the recursive SOM and other extensions under a comparative aspect. Eventually, we introduce the hybridization idea of the map SOM with genetic algorithm GA, by making it applyable to SOM recursive model 'RecSOM' in an attempt to reach the natural evolution of the human thought as regards to recognition.

In this overview, we are interested in the study of various approaches introducing temporal dimension to ensure dynamic representations generalized by the map SOM. The integration of time can be made implicitly even the case of TOM, ST Kohonen or explicitly by recurrent loops even the case of RSOM, CSOM, and RecSOM. Elementary knowledge concerning the data representation techniques by maps SOM is necessary to understand a major part of the problems presented.

We start with a discussion concerning the interest of maps SOM at level of section 2 and the models being based on the idea of `laky integrators' to introduce time on the level of section 3 and 4.

In section 5 we deploy the principal models of dynamic SOM.

The last section brings the possibility of a hybridization of these temporal models by GA, the performance indexes used and the suitable operators of evolution.

2. Kohonen model: the map SOM

2.1. General aspect

The algorithm of kohonen organizing maps (95; 97 and 2001), is a nonlinear algorithm of projection and an algorithm of classification at the same time. It is characterized by the representation of static data: the output depends only on the current states of input. By adapting the map to the statistical distribution of its entry space we carry out a classification/ categorization of these entries into a monodimensional or two-dimensional map according to the winner takes all (WTA) principle illustrated by the normal curve or according to the Mexican hat principle[1], [5], [8], [19], [20], [27],(see fig.1). Following the activation of the network with a data vector, the algorithm will be associated with these data, a set of prototypes organized at the neuronal level according to a low dimension structure (1D or 2D). Each prototype represents a subset of original data which we can consider as a class while having preserved their topology.

Conceptually, the principle of algorithm SOM is, just like for Competitive Learning, to repeatedly move prototypes within a distribution of data contained in a unit D. At the end of the execution of algorithm SOM, the distribution of the prototypes represents an approximation of the initial distribution of the data. Algorithm SOM thus makes it possible to quantify data in vectored manner. However, the SOM has a particular characteristic which is the use of a grid connecting the various units between them in a topology included by entry space.

The SOM algorithm is defined as follow: in the training phase, the data of unit D are successively presented to the algorithm, iteration after iteration.

As for Competitive Learning, in each iteration, the current data x(i) is selected according to a considered distance of measurement.

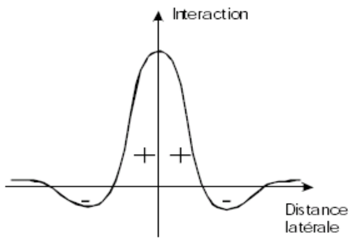

Figure1. Mexican hat describing the side influence of the victorious neuron on its neighbors in a competitive SOM.

In other circumstances, we note that the popularity of the SOM is considerable. It is enough to re-examine these thousands of important practical applications in various fields and in the field which we are interested [30], [49], [54], [55]. Either, we note that the Kohonen algorithm is quite, since 1982, the most used unsupervised algorithms of classification. The objective of his unsupervised learning is to represent a large set of stimuli faithfully by a set of neurons (prototypes).

Finally, it should be announced that for the construction of its representations, the kohonen algorithm uses an iterative mode.

2.2. Functional principle

Each neuron i of the map compares its weight vector W_i with a static data vector of input x(t); with t which is the time corresponding to an iteration. The quantification error associated with neuron i is given by the following Euclidean distance:

$$E_i = \|x(t) - w_i\| \quad \text{Eq. (1)}$$

Each static data vector of entry is applied simultaneously to each neuron of the grid. And for an entry vector given, the winner neuron "v" is the unit which minimizes this quantification error from where we have:

$$E_v = \min E_i \quad ;i \in N \quad \text{Eq. (2)}$$

The training rule updates the neurons weights which belong to a vicinity "neighbor" of the winner neuron, by bringing them closer to the entry vector:

$$\Delta w_i = \gamma . h_{iv}(x(t) - w_i) \quad \text{Eq. (3)}$$

γ is a training rate and h_{iv} is a neighbor function, which decreases by the distance between units i and v on the map. The vectorial quantification which characterizes the algorithm SOM consists in forming an inexpensive representation of a stochastic entry vector by projection of an entry space of dimension N in output space (the map) with dimension 1 or 2. To evaluate the SOM quality as a quantifier, we determine the error of quadratic quantification average, definite as the hope of this error associated to the winner neuron [6]:

$$E_v^2 = \prec \|x(t) - w_v\|^2 \succ \quad \text{Eq. (4)}$$

With < . > indicate the statistical hope.

Because the input is mapped onto a discrete, usually lower dimension output space, the SOM is typically used as a vector quantization (VQ) algorithm. The weights of the winning node are the vector quantized representation of the input [56], [60], [69].

The SOM algorithm can be described as follows:

Procedure train_SOM
Begin
Randomize weights for all neurons
For (i = 1 to iteration number) **do**
  Begin
  Take one random input data vector (e.g. a phoneme),
  Find the winning neuron (using Eq. (2)),
  Find neighbours of the winner (with Mexican hat),
  Modify synaptic weights of these neurons (by Eq. (3)),
  Reduce the γ and h_{iv}
  End

End

There have been many attempts at integrating temporal information into the SOM. One major technique is to add temporal information to the input of the SOM. For example, exponential averaging and tapped delay lines were tested in 1990 and 1991 by Kan. Another common method is to use layered or hierarchical SOMs where a second map tries to capture the spatial dynamics of the input moving through the first map [37], [58].

Recently, researchers have begun integrating memory inside the SOM, typically with exponentially decaying memory traces. Privitera and Morasso have created a SOM with leaky integrators and thresholds at each node which activate only after the pattern has been stable in an area of the map for a certain amount of time. After that, the leaky integrators idea was developed by Chappell and Taylor in TKM model. It is considered as the first step to reach the biology of human brain.

The principal idea to introduce time into the map SOM in order to instigate there, while approaching the biology of the human brain, was posed by Chappell and Taylor in 1993 when they noted that the weight of each neuron represents a state of memory of input data information . And that this active memory (weight) loses this data gradually while following a law of exponential discharge, which makes it possible to model each neuron of the map SOM by an active filter passes low first order said integrating or "leaky integrator".

Their model of TKM thus reasons on the modeling of each neuron by an integrator placed at the exit. This idea have license to introduce temporal dimension into the map SOM.

In 1998 Varsta noted that model TKM cannot represent all the entry sequences owing to the fact that the exits converge towards linear combinations of the entries. To cure this risks, it made a light modification on TKM by modelling each neuron of the map SOM by an integrator placed at the neuron entry and not to its exit. This modification thus consists to a simple slip of the integrator position from a neuron exit towards its entry: it is the model of the recurrent SOM (RSOM).

3. The temporal kohonen map: TKM

This map represents one of the first attempts to integrate temporal information in SOM by Chappell and Taylor in 1993. It is considered as an interesting unsupervised approach for TSP (Temporal sequence processing), which derived from the Kohonen's Self-Organizing Map algorithm. It is trained by the same training rule of normal SOM, except that the activity of a TKM unit is defined as a function of the last entry vectors [2], [3]. The TKM model supports essentially on the neuron weights modification of the kohonen map to allow the network introducing temporal dimension.This approach is built on the biological neuron modeling with a simple realistic mathematical model describing the change of potential inside a neuron and thus the behavior of this one by the use of temporal differential equations. Each unit i will be carried to a potential $v_i(t)$ at time t, then it will be updated according to the following rule [3], [5], [6]:

$$V_i(t) = d.V_i(t-1) - (\frac{1}{2}).\|x(t) - w_i(t)\|^2 \qquad \text{Eq. (5)}$$

W_i is the reference `code book' or the vector weight associated with the unit i And, d is a time-constant ranging from (o) to (1) which indicates a term of memory lapse. $V_i(t)$ is the unit i activation at time t, x(t) is the entry sequence. However, the best matching unit BMU in the TKM, is that maximizes V_i [5], [9]:

$$BMU(t) = \arg\max V_i(t) \qquad \text{Eq. (6)}$$

These differential equations model the neurons by low frequency linear filters known as the "leaky intégrators", baring in mind that a differential equation operates a filter and the order of the equation gives the order of the filter. We deduce that the insides activity of a map neuron is obtained by this differential equation binding the entry and the current activity. The neuron output was then calculated compared to this inside activity. Thus, this model makes possible to leave a trace of whatever activity, this implies that the traces of the last activities can coexist among to this remanent inside activity of the membrane and being able to interact by training. The TKM is intended not only for the entry sequences distinction, by the choice of the best unit adapted for each sequence, but also for the creation of sequences contexts [4], [5]. It differs from SOM only in its output activities. We note that the SOM outputs will be initialized to zero after the presentation of each entry vector and the selection of the BMU. In the TKM these outputs are replaced with the leaky integrators considered as low frequency linear filters which, once activated, they lose their activity gradually from where the name is derived. It has been shown by Koskela and al., 1998 in that this algorithm is strongly limited, due to the fact that the weights converge towards linear combinations of the entry vectors that limits the possible representations [3], [4], [7].

An alternative to converting temporal information into a static form is to use low-pass filters, such as those formed by resistor-capacitor electrical circuits, which can be added to either the inputs or the outputs of the neurons. And then such dynamic units are usually called leaky integrator neurons. This idea was introduced in an unsupervised context by Taylor, Plamondon and Privitera (1995). Kremer (2001) has named this type of short term memory mechanism Feedforward Exponential Decay 'FED' memory. Then, the variant of SOM called TKM uses FED memory[26], [35], [45].

The "leaky integrators" make TKM possible to take account of the vectors history of entry considered previously during the calculation of the distance between the current vector and the various prototypes. This calculation of distance is in fact a balanced sum of the distances between the preceding vectors and the considered prototype, for each neuron. Weighting follows a decreasing exponential law. As one moves away in the past, the choice of the prototype being a more functional of some last vectors which have just been presented to the algorithm.

The balanced sum thus corresponds to an integration of the direction of the change from one to another iteration of the algorithm. This approach based on "leaky integrators", coupled to hierarchical SOM, is applied for example in speech recognition.

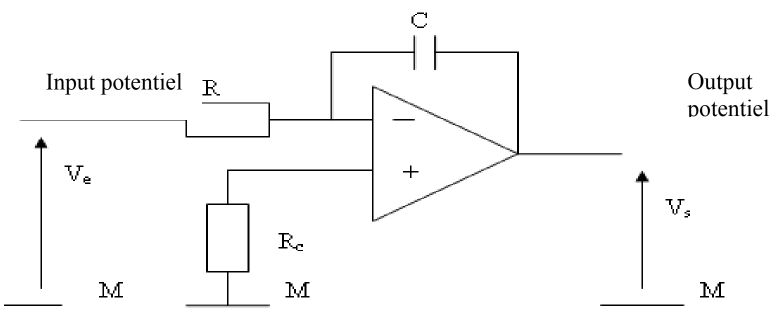

Figure 2. Modeling of neurons exits of the map SOM by a low pass filter electric

The constant-time of the RC circuit is equal to product R×C expressed in Second. This circuit of filtering added with an operational amplifier is called integrator, in the fact, it introduces the factor time by effect of load and discharge: t=R.C. The potential of discharge will thus be done gradually while following an exponential law which depends on the values of R and C. This modelling is inspired by the biology of a neuron unit; using its resistance in front of data flow and its capacity being able to support this flow in the form of a potential. We suppose that the condenser is initially discharged: Uc (t=0) = 0 Volt.

The mathematical equation of the discharge of the condenser initially charged with E will be the following one:

$$U_c\ (t) = E.e^{-t/RC} \qquad \text{Eq. (7)}$$

4. The Recurrent SOM

Some of the problems related to original TKM have an adequate solution by moving only the leaky integrators from the units outputs towards their entry. This gives rise to a modified TKM called `Recurrent Self Organizing Map', RSOM, proposed by Varsta and al.. 1997. The fact of moving the leaky integrators from the outputs towards the entries of each unit of the map, ledes a temporal vector of leaky difference given by the following relation:

$$y_i(t) = (1-\alpha).y_i(t-1) + \alpha.(x(t) - w_i(t)) \qquad \text{Eq. (8)}$$

Where α lies between (0 and 1), indicating a coefficient of leaky which replaces d for the TKM. A unit of RSOM will thus be schematized as follows:

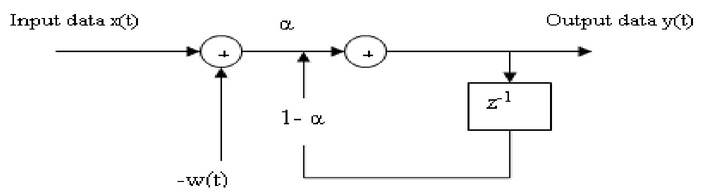

Figure3. Representation of a unit of RSOM which acts that recurrent filter

That implies a high value of α corresponds to a short memorizing, however a low value of α corresponds to a long memorizing and a slow weakening of activation. If (α = 1), RSOM is transformed into normal SOM refering to y_i (t) equation [8], [9], [42], [47]. Obviously, the idea to take account of the last execution of the algorithm, in a recursive way, was used in Recurrent SOM (RSOM). The principal difference between the RSOM and the TKM is that the TKM integrates the direction of the activity change only during the distance computation, whereas the RSOM also integrates this change during the position modification of the prototype. The prototype in addition preserving this history of the changes for the after comparisons. This idea is translated, in this model, by a vector of leaky difference associated with each neuron i and defined according to Eq. (8). So the model of RSOM will be more coherent with the algorithm of standard SOM. Finally, it is necessary to recall that RSOM was evaluated according to a Synthetic data in the form of random sequences of symbols with an alphabet limited and noise additive [3], [61], [68], [70].. This model is different from TKM in that TKM associates all the temporal processing capacity with the trajectory of the best matching units. But, RSOM explicitly includes recurrent connectivity into the neural output. This results in RSOM models having the capacity to explicitly capture temporal patterns in the original input, whereas the TKM concentrates on best matching unit sequence learning as represented in follow:

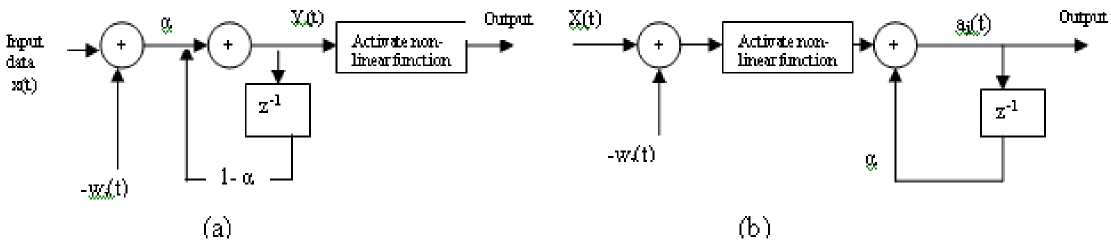

Figure4. Representation of a RSOM unit in (a) and a TKM unit in (b).

The main difference between the RSOM and the TKM is that the TKM integrates the direction of the change only during the calculation of distance, whereas the RSOM also integrates this change during the modification of the position of the prototype, the prototype in addition preserving this history of the changes for the later comparisons.

5. The Recursive SOM

Another approach is the Recursive SOM (RecSOM). This model represents an application, on the kohonen unsupervised algorithm SOM, of a self- referent training class proposed by Thomas Voegtlin,(2002), deduced starting from the SRN (simple recurrent network) of Elman in 1990 which acts of a training by prediction thus supervised. The SRN is a perceptron modified by a hidden layer, using a delayed copy of the activities of its layer hidden like additional entry. Its task is to learn from associations of output-input sequences. It is involved with the algorithm of retropropagation of error (see fig.5):

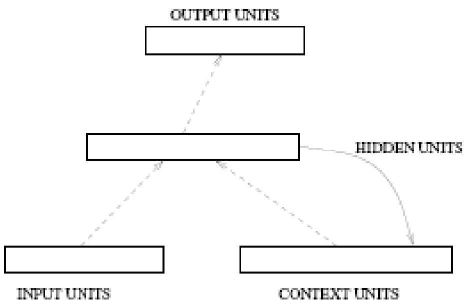

Figure5. Principle of SRN with loop of return introducing the idea of self reference

The representation in the context layer of a SRN is the same as in its hidden layer. Consequently, it is said that the hidden layer learns how to represent its own passed activities, since we will make it introduces each time its lasts activities. In this direction, the representation in this hidden layer is known as "self referent" [6], [11]. This self reference thus will influence the training while acting on the function of error between desired output and real output located at a well defined level of iteration. It represents a property of the algorithm of training and not of architecture. What is appropriate to say that the self reference thus implies that convergence by minimizing the error is a problem of moving target; it tends to optimize a subjective and not objective function of error.

This idea is applied to SOM thus giving the recursive SOM whose training is self referent with the direction that the map learns how to classify its own last activities. In the same way, RecSOM is iterative by applying the unsupervised algorithm of kohonen. The principle is illustrated by (fig.4) [6], [51], [32]:

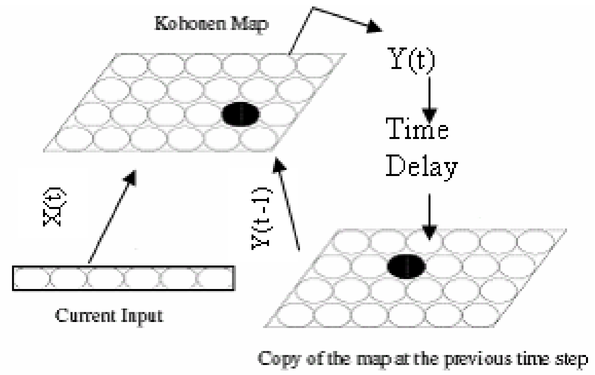

Figure6. Principle of recursive SOM with loop of return introducing the idea of self reference

The recursive SOM is thus a SOM with self reference; this results in the addition of recurrent connections to original architecture of SOM of a manner which is compatible with it self-organisation's principle [1], [6]. Connections associated with the context and the layer of entry are homogeneous (they are defined by the same equations of neuronal activity). The entry and the copy of the last activities are regarded as only one vector of entry for algorithm SOM. Each neuron will be involved to learn how to represent a couple (entered, context). Thus, from the representations of long sequences are learned repeatedly, starting from the representations already learned from shorter sequences.
The error of quantification will thus be given by the following relation [2], [8]:

$$E_i = \alpha.\left\| x(t) - w_i^x \right\|^2 + \beta.\left\| y(t-1) - w_i^y \right\|^2 \qquad \text{Eq. (9)}$$

The network learns by associating the current entry to the preceding states from activity. Consequently, each neuron becomes sensitive to a sequence of entry, and the BMU is given by:

$$V = \arg\min\{E_i\}; i \in N \qquad \text{Eq. (10)}$$

The rules of training used to update the weights feed-forward and recurrent are given by:

$$\Delta w_i^x = \gamma.h_{iv}.(x(t) - w_i^x) \qquad \text{Eq. (11)}$$

$$\Delta w_i^y = \gamma.h_{iv}.(y(t-1) - w_i^y) \qquad \text{Eq. (12)}$$

The dynamic aspect of the recurrent connections makes it possible to have unstable representations. Confines to this, a promising idea consists in applying a transfer function F of the exponential type, selected in an empirical way so that it is continuous and having values ranging between (0 and 1):

$$y_i(t) = F(E_i) = \exp(-\alpha.\left\|x(t) - w_i^x\right\|^2 - \beta.\left\|y(t-1) - w_i^y\right\|^2) \quad \text{Eq. (13)}$$

The idea to have a transfer function to stabilize the training of recursive SOM rises from its modeling by a network with only one neuron whose output Y is defined by the transfer function of a sum entry weight.

The algorithm of self reference is validated by Thomas Voegtlin in 2002 on a two-dimensional recursive SOM of size 20x20 neurons, involved on an English text; the novel ` Brave man new world' of Aldous Huxley. Each letter is encoded on 5 bits then presented at the network a letter by a letter. The symbols of punctuation are removed text and the neuronal activities were given to zero between two words. The rate of training was constant: $\gamma = 0,1$. The function of vicinity had a constant size: $\sigma = 0,5$. The other parameters was selected: $\alpha = 3$ and $\beta = 0,7$.

6. Other extensions

Temporal data processing is a very important task, to which there is no unified approach . In the algorithm of RecSOM, each prototype has a vector weight representing its position in the distribution, and has also another vector representing the context of activation of all the map at the time of the preceding iteration. The selection of the nearest prototype is based in this case at a distance taking account on the one hand of the difference between the data and the prototype weight, on the other hand of the difference between the preceding context and the context of the prototype. Updated prototype requires the modification of the weight and the context of the winner prototype with its neighbors [1], [6], [7], [25]. On the basis of this principle, we can affirm that Contextual SOM (**CSOM**) developed by voegtlin T. in 2000, is in fact the same algorithm as RecSOM.

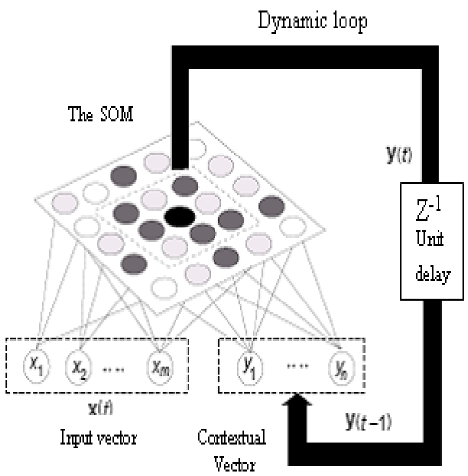

Figure7. Principle of the contextual sel-organizing map: CSOM with loop of return introducing the temporal context.

Due to the diversity of properties of such temporal signals. If the properties of signals do not vary with time, the signal is said to be stationary. However, many real signals are generated by systems that have time varying parameters for exemple the continious speech, and thus they are called non stationary. In such case, an important tool used is the windowing or the

segmentation, whose goal is to find regions where the signal properties are reasonably stationary. Some TSOM have been proposed with the aim of segmenting input sequences.
An other new SOM paradigm is the adaptive-subspace self organizing map **ASSOM** [38], [40], [41]., has been proposed in litterature to solve a long-standing problem in the theory of perception: the formation of perceptual invariances, such as the ability to recognize words independently of the speaker, or objects independently of orientation, and their emergence in learning processes. A neuron in the ASSOM represents a linear subspace which in turn represents different invariant features. In this case, matching means comparison of orthogonal projections of the input vector x(t) on the different subspaces. Then, the ASSOM uses the concept of a representative winner for a set of x(t) vectors that occur adjacent in time called the episode. Hence, the various map units adaptively develop into filters of invariant features . The ASSOM model has been successfully applied to a variety of speech recognition tasks and it is'nt originally conceived as a technique for temporal sequence processing only. It is used for recognition independently of certain transformations.

The algorithm Merge SOM (**MSOM**) recently developped by Barbara Hammer and Marc Strickert, October 2004 is very close to the RecSOM algorithm. In this algorithm, the prototypes also have a weight vector and a context vector [43], [29]. The principal difference lies in the definition of the context vector. In general, the merge SOM context refers to a fusion of two properties characterizing the previous winner: the weight and the context of the last winner neuron are merged by a weighted linear combination. During MSOM training, this context descriptor is kept up-to-date and it is the target for the folowing context vector of the winner neuron and its neighbourhood. The temporal context of MSOM combines the currently presented pattern with the sequence history in an intuitive way by referring to a merged form of the winner neuron's properties [12], [14]. So that, MSOM accounts for the temporal context by an explicit vector attached to each neuron which stores the preferred context of this neuron. The way in which the context is represented is crucial for the result, since the representation determines the induced similarity measure of sequences.
In RecSOM, the vector of context represents the activation of various prototypes of the map at the time of the preceding iteration. However, in the MSOM, the context consists of a linear combination between the weight and the context in the preceding iteration. The context is thus richer, but also more complex. The MSOM combines a noise-tolerant learning architecture which implements a compact back-reference to the previous winner with separately controllable contribution of the current input and the past with arbitrary lattice topologies. With respect to representation, MSOM can be interpreted as an alternative implementation of the encoding scheme of TKM, but MSOM possesses larger flexibility and capacity due to the explicit representation of context: In MSOM neurons specialize on both input data and previous winners in such a way that neuron activation orders become established; thus, order is coded by a recursive self-superposition of already trained neurons and the current input.

The temporal organization map (**TOM**) explored by S. Durand and F. Alexander in "TOM, A New Temporal Neural Net Architecture for Speech Signal Processing, in proceedings of ICASSP, 96" integrates a cortical column model, SOM learning and separate temporal links to create a temporal Kohonen map [10], [21], [39], [41]. The TOM is split into super-units that are trained via the SOM learning algorithm. Winning units from each super-unit activate and then decay. Temporal links are made between the currently firing node and any node which has an activity above a threshold. Thus there can be multiple links created for each activation, allowing for the pattern to skip states. Wiemer, in 2003, has developed even more the TOM model by giving an influence to temporal information in the process of the self organization map SOM.

The contribution of TOM compared to SOM is the transfer of the temporal distance between stimuli in a space distance. This transfer is ensured by the phenomenon of the dynamic propagation of neuronal activations in the waves form allowing the signals current and passed to interact space-time lies in the map SOM. The TOM tests to break up, like SOM, the training in a sequence of discrete stages.

The **SARDNET** architecture created by James and miikkulainen in 1995 has proved successful in learning and recognizing arbitrary sequences of binary and real numbers, as well as sequences of phonemic representations for English words [7], [22], [34]. The SARDNET extends the Kohonen Feature Map architecture with activation retention and decay in order to create unique distributed response patterns for different sequences.

Its principle consists to activate units on the map at the same time by a sequence of input vectors. The past winners are excluded from further competition, and their activation is decayed gradually to indicate position in the sequence. It is a simple TSOM used to detect no-vectorial forms of data, such as sequences. This model, called SARDNET abbreviate from 'Sequential Activation Retention and Decay Network' includes a simple retention and decay mechanism with the aim to reform a unique set of activated neurons for each distinct input sequence. It adds exponential decays to each neuron for use in the detection of node activation sequences. Once a node activates for a particular sequence, it is not allowed to activate again. Therefore, at the end of the sequence presentation, the sequence of node activated can be detected or recreated using the decayed outputs of the SOM. The exponential decay, however, provides poor resolution at high depths and thus will perform poorly with noisy and long sequences. So, in addition to the two first steps of SOM training, a third step assigns the value 1 to the activation of the BMU and excludes it from subsequent competitions. The fourth step updates the activation $a_i(t)$ of all units in the map as follows:

$$a_i(t+1) = \alpha.\ a_i(t) \qquad \text{Eq. (14)}$$

where $0< \alpha <1$ is a decay parameter and $a_i(0)=0$ for all units; At the end of a sequence presentation, the neuron with highest activation represents the most recent input vector and the one with the lowest activation represents the first vector of the input sequence.

Eventually we remark that the validation of SARDNET is effected on phonemic word representations obtained from the CELEX database of the Max Planck Institute for Psycholinguistics and converted into International Phonetic Alphabet (IPA)-compliant representation, which better describes similarities among the phonemes. The words vary from five to twelve phonemes in length. Each phoneme is represented by five values: place, manner, sound, chromacity and sonority.

Fancourt and Principe have proposed a **TSOM** for the identification and segmentation of piecewise stationary series [27], [33], [50], [64], [65], [66]. In this model, the concepts of leaky integrator neurons and operator maps are used. Such neuron is a linear predictor operator trained with the LMS algorithm. The wining predictor is found as follows:

$$BMU\ (t) = arg\ ming\ \{a_i(\ t)\} \qquad \text{Eq. (15)}$$

$a_i(\ t)$ is computed according to the concept of leaky integration:

$$a_i(t) = (1-\alpha).\, a_i(t-1) + \alpha .e_i(t)^2 \quad \text{Eq. (16)}$$

$0<\alpha<1$ is the memory parameter and e_i is the prediction error. It can be deduced that in the SOM with Sequential Activation, Retention, and Delay (SARDNET), the idea of the recursive retention of information is supplemented by a temporary mechanism of desactivation of the prototypes, which cannot be any more modified during a certain time after an adaptation.

Else**, SOMSD** has been proposed for general tree structures by M. Hagenbuchner, A. Sperduti, and A. Tsoi, in “A self-organizing map for adaptive processing of structured data”, 2003, then by Stephan Bloehdorn and Sebastian Blohm, Institute AIFB, University of Karlsruhe, Germany in “A Self Organizing Map for Relation Extraction from Wikipedia using Structured Data Representations’. The general idea is to derive automatically sets of documents, for example articles, from Wikipedia [13], [15] [29], [52], [63]. The SOM for structured data (SOMSD) is restricted to regular lattice structures. As for RecSOM, an additional context vector is used for each neuron, but only the last winner index is stored as lightweight information about the previous map state. During training, the winner and its neighbourhood adapt weights and contexts of input in step t and the winner in step (t − 1). Thereby, grid topology and the indices of neurons are expressed as elements of a real-vector space. Obviously, the context of SOMSD relies on ordered addressing in terms of the topology of the neural map; consequently, it cannot be combined with alternative models such as neural gas. If complex sequences are mapped to low-dimensional standard Euclidean lattices, topological mismatches can be expected.

The Self Organizing Maps for Structured Data: SOMSD is an extension to the SOM for scenarios where information is not only contained in the individual patterns but also in the order in which they are presented. The order of the presentation can be used to encode sequences or paths in a directed graph on the input data. The SOMSD allows taking the context of a presented input pattern into account. This is done by means of a recursive formula over the entries of a given sequences during winner selection, a characteristic shared with other recursive SOM models. However, in contrast to other recursive SOM approaches as e.g. the Temporal Kohonen Map (TKM) proposed in [3], [5], [26], [27], the SOMSD represents the sequence context only by the location of the winner neuron of the previous sequence item.

We note that the SOMSD was proposed for structures of tree, by modifying the measurement of distance used in traditional algorithm SOM. Thus, the algorithm SOMSD can be used in the temporal context and becomes algorithm SOM for Sequences, **SOM-S**. In the same way, the extension of the SOMSD has topologies of more general grids with triangular vicinities, such as defines in Hyperbolic SOM (HSOM), led to the **HSOM-S**.

The U-Matrix Clustering for unified distance matrix is a SOM-based clustering technique that exploits the fact that when parts of the input space X are mapped onto the SOM, the area of the map representation correlates with the density of data samples from that part independent of the corresponding volume in X. The U-matrix allows detecting such differences in density by assigning each node the sum of the distances of its weight vector to those of its direct lattice neighbours.

Actually, the large space which occupied by Internet and mail services in individuals life today, their influences on the economic transformations and the international relations contributed to the birth of the **WEBSOM** project.

A certain number of works were interested in the topology of the neurons vicinity, which is a principal limitation. Following the project WEBSOM of information extraction (Data-Mining in English) for a vast documentation (texts) undertaken recently by Kohonen and al., the researchers put the following suggestion: the choice of rectangular topology is it the best two-dimensional representation of a complex input space of high dimension? Ritter and al. bring their brief reply with a concrete algorithm: **HSOM** [23], [57]. The map of neurons employed consists of a projection of a hyperbolic space on R^2. Ritter concluded with a better representation for a space of entry to high dimension, but especially to the possibility of forming a hierarchical structure in the map of neurons. One of the disadvantages of this approach is that the topology of the map is always specified a priori.

HSOM-S transfers the idea of SOMSD to more general grids expressed by a triangulation of a two-dimensional manifold, such as a hyperbolic plane, in order to match better the data topology of sequences. The previous winner, adjusted through Hebbian learning, is referred to by a coordinate within a triangle of adjacent neurons. Such an approach constitutes a step towards more general lattice structures which are better suited for the presentation of input sequences but it is still limited to a first fixed topology. Obviously, RecSOM offers the richest notion of history because temporal context is represented by the activation of the entire map in the previous time step; however, RecSOM also constitutes the computationally most demanding model. SOMSD and HSOM-S still use global map information, but in a compressed form; their storage of only the location of the winner is much more efficient and noise-tolerant, albeit somehow lossy, in comparison to the whole activity profile. In both models, context similarity, expressed by the comparison of distances between grid locations, depends on the priorly chosen static grid topology. SOMSD and HSOM-S compress the costly RecSOM activation profile to only the topological index of the last winner in the neuron grid [13], [15], [36].

In other case, an approach for abstracting invariant classifications of spatio- temporal patterns was presented in a high dimensionality applying an early proof-of-concept to shift and scale invariant shape recognition called Hierarchical Quilted Self-Organizing Map (**HQSOM**). It was developed by Jeffrey W. Miller and Peter H. Lommel; Draper Laboratory, 555 Technology Square, Cambridge- USA, 2004, using recurrent self-organizing maps (RSOM) arranged in a pyramidal hierarchy, attempting to mimic the parallel-hierarchical pattern of isocortical processing in the brain [16]. The HQSOM is an implementation of such architecture, using a single algorithm, the RSOM, applied in masse. Note that HQSOM is very different than the Hierarchical SOM, HSOM, which consists of SOMs arranged serially, rather than in parallel, and does not involve time. In the HQSOM, the input is parsed into overlapping receptive fields, each of which is connected to a SOM-RSOM pair in the first layer. Each layer consists of a "quilt" of SOM-RSOM pairs having overlapping receptive fields.

Actually, Growing Recurrent Self Organizing Map (**GRSOM**) is proposed by Ozge Yeloglu, A. Nur Zincir-Heywood and Malcolm I. Heywood in Faculty of Computer Science, Dalhousie University-Canada, March 16, 2007. It is embedded into a standard Self-Organizing Map (SOM) hierarchy. That's why, the KDD benchmark dataset from the International Knowledge Discovery and Data Mining Tools Competition is employed. This data set consists of 500,000 training patterns and 41 features for each pattern. While RSOM is a promising structure in temporal sequence processing area, it is difficult to decide an appropriate network structure for a given problem. Since a fixed network structure is used in terms of number and arrangement of neurons, which has to be defined prior to training, this often leads to a significant degree of trial and error when deploying the model. Therefore, previously proposed growing neural network methods [17], [23], [24], [59] motivate the idea of a Growing Recurrent Self-Organizing Map (GRSOM). The target is to design a RSOM model that determines the number and arrangement of units during the unsupervised training process. A growing behaviour is added to the RSOM. Training of this model starts with only one neuron and new neurons are added until the desired accuracy is reached. For every unit that is newly added, the weight and the leaked difference vectors are initialized to zero. Then, an input from the training data is presented to the new neuron and weight and leaked difference vector values are adapted. The next input is presented to the neuron until the leaked difference value of the neuron is under a threshold, which is decided at the beginning of training. If the number of neurons is less than the desired maximum number of neurons, then a new neuron is added to the network until the end of the training data is reached.

The GISTSOM, for graphical interface system TSOM, is a software package developed by the authors for the definition and comparison of space-time trajectories in multi-parameter, spatio-temporal data.

The standard Kohonen mapping algorithm described above is applied to sample data collected at one point in time. However, it can be repeated on data existing across several time periods, for which an extension GISTSOM has specifically been developed. In GISTSOM, the sample data weight vectors used in the training of the initialized SOM is taken randomly from across all of the time periods for which data are available. In this way, the weight vectors against which the Kohonen map learns implicitly include a temporal dimension, and a spatiotemporal Kohonen map evolves.

In the case of GISTSOM a simple coordinate tolerance measure has been used as a measure of similarity. Sample data with similar space-time trajectories can then be identified as samples whose multi-dimensional values have changed in a similar manner through time.

The self- organization by balanced excitatory and inhibitory input model (**SOBEII**) developed, more recently, in 'A Novel Self-Organizing Network Determines The Proper Cluster Structure Automatically' by Bin Tang, Malcolm Heywood, Michael Shepherd at Dalhousie University, Faculty of Computer Science Halifax, Canada, using balanced excitation and inhibition and anti-Hebbian learning strategy, SOBEII is capable of automatically determining the proper cluster structure of given datasets in a robust manner. This model is demonstrated using both synthetic and real datasets.

The STAN was developed by G.Vaucher in 1996. It can be regarded as a model approaching the traditional artificial neuron. Indeed the STAN has the algebraic properties of the traditional neurons, by adding to it the capacity to treat asynchronous data. The principal idea of STAN is to equip the artificial neuron with a capacity to integrate at the same time temporal and space information in its calculations. Its principle is to code discrete events on two degrees of freedom, amplitude and date, in the form of complex numbers with also two degrees of freedom: amplitude and phase. The technique consisting in adding delays to the inputs level to introduce temporal information as that is made in certain models of traditional neurons and the mechanism of leaky memorizing (leaky integrators) of neurobiologic inspiration are gathered in a single model of neuron. Taking in consideration the temporal and space presents advantages. It becomes possible to make the difference between a great event in a remote past and a small recent event what is not the case with the basic neuron. Remain to stress that coding ST without asynchronous dynamics such that it was put apply by N.Mozayyani comes under the field of the traditional ANN. This coding static ST indeed makes it possible to jointly process the data space and temporal one contained in the data but, once a sequence is coded in form ST, the entry vector which results from this is fixed and its treatment concerns traditional ANN adapted to the body of the complexes. Preceding architectures of STANN had not been conceived to take into account the potential asynchronous dynamics of the STAN. Baig in 2006 introduced the idea of an application in real-time of coding ST (space-time) with autonomous neuronal units treating asynchronous impulses such as St-kohonen. The STAN is derived from the model of W. Rall of the biological neuron. Its interest main is its capacity to integrate at the same time space information and temporal information in its calculations what gives him the possibility of detecting asynchronous sequences of events.

Euliano and Principe proposed in 1995 and 1999 the self organizing map with temporal activity diffusion: **SOMTAD.** A model based on the biologically inspired diffusion of activation through time over the neurons in the map and temporal decay of their activations. This Algorithm also uses the “leaky integrators” by addition the diffusion memory of activity through the structure of the SOM [44], [48].

The basic concept in the SOMTAD is the activity diffusion through the output space. The activation of a unit in the map network causes activity to diffuse through the network and affects both the training and the operation of the map. The activity diffusion moves, in general, through the lattice of a SOM structure. When the activity diffusion spreads to neighbouring units, the thresholds of these neighbouring units are lowered, creating then a situation where the neighbouring units are more likely to activate next. An other temporal functionality included in SOMTAD architectures is the decay of output activation over time: “FED memory”. That means when a unit becomes active, it maintains a portion said: “exponentially decaying” of its activity after it becomes active.

Another concrete algorithm, the **TS-SOM** (Tree Structured Map Coil-Organizing), proposed by Koikkalainen, is based on geometry of tree in growth. In this algorithm, there is progressive specification of the scale of training, as neurons are added. The structure of tree produces categories and under categories, modelling at exit east thus treated on a hierarchical basis: we can read the map at various levels of tree structure. This approach opens the way of the constructive algorithms, in which the number of neurons increases with the wire of the training. We can quote the most known of them the **GSOM**, or the **GNG** (Growing Neural Gas), his performances being extremely good.

In the GNG, the neurons are added to the network with a regular frequency. To know where to add a new neuron, basing oneself on a function of error allotted to each neuron. The larger of this one is, the more area concerned with need for neuronal reinforcement. With this algorithm, the problem to fix a topology by advance is solved and the error of modelling is very weak. The intermediate neurons disappear from the final map, topology is learned automatically. The problem of this approach is obviously that it makes total abstraction of biological reality!

Contrary, certain research teams raise the question of biological modelling, without particular consideration for the developments in artificial intelligence. One of the most elaborate models is the **RF-LISSOM** (Receptive-Field Laterally Interconnected Synergistically Map Coil-Organizing), whose results rather accurately reproduce the experimental observations in the primary education visual cortex.

In this model, the dynamics of the neurons is governed by a system of differential equations, the neurons being of type `summation and activation' (Integrate-and-Fire). The training is unsupervised by hebbien type (characteristic of the SOM), and the connectivity of the network is dynamic because useless connections progressively are removed. All the ingredients are thus joined together the most accurately to reproduce biological behaviour of the neurons with knowledge and technique we have today.

Model of Kangas: A two layer model with an integrating memory connecting the maps was proposed by Kangas in1994. His approach concerns a hierarchical model employing two maps and an integrating memory between them. The first map of SOM performs (produces) a non linear transformation on the input data. The transformations are stored in integrating memory and subsequently serve as the input for the second map. When broken down they essentially have an RSOM in the bottom with a basic SOM on the top [12], [31]. The integrating memory between the maps is realized with leaky integrators like in the TKM model. The transformation values *yi* stored in the memory are inverted normalized Euclidean distances between data and code-book vectors such that the BMU gets equal to 1 and the other units smaller values accordingly, thus for *yi* we can write:

$$Y_i\,(t) = \left\| X\,(t) - W_{b\,(x)} \right\| \,/\, \left\| X\,(t) - W_i \right\| \qquad \text{Eq. (17)}$$

The obtained activity values are sharpened by squaring them a few times. In these models the outputs of the first map are stored in an integrating memory to provide the input for the second map.

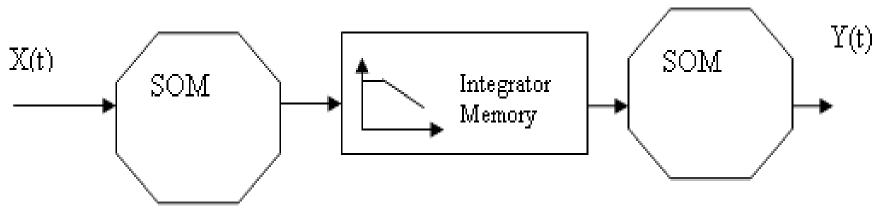

Figure8. Principle of a hierarchical model employing two maps and an integrating memory (Kangas, 1994).

Guilherme A. Barreto, Department of Teleinformatics Engineering, Federal University of Cear´a-Brazil purposes to apply the **SOM** to **TSP** problems, including curiosity about or familiarity with the algorithm, which can surely take the following two points for granted:

- Local Nature of SOM-based Models - Roughly speaking, SOM-based models for time series prediction belong to the class of models performing local function approximation. Thus, SOM-based TSP models allow the user to better understand the dynamics of the underlying process that generates the time series, at least within the localized region used to compute the current model's output. This localized nature of SOM-based model combined with its topology-preserving property are useful, for example, if one is interested in time series segmentation or time series visualization problems.
- Simple Growing Architectures - As one could expect, clustering of the input space is an essential step in designing efficient SOM-based TSP models. A critical issue in the process of partitioning the input space for the purpose of time series prediction is to obtain an appropriate estimation of the number of prototype vectors. Over- or under-estimation of this quantity can cause, respectively, the appearance of clusters with

very few pattern vectors that are insufficient to building a local model for them and clusters containing patterns from regions with different dynamics.

KSOM model: Using Time-Varying Local AR Models from Prototypes, the last SOM-based model for time series prediction that we describe was introduced by Barreto and al, being called the KSOM model. The KSOM model combines the vector quantization approach of the **VQTAM** model with that of building local linear AR models. This approach uses the well-known K-means clustering algorithm to partition data vectors for building local AR models. With (K > 1) are the closest prototypes.

However, instead of q time-invariant local linear models, the KSOM works with a single time-variant linear AR model whose coefficient vector is recomputed at every time step t, directly from a subset of K (K ≪ q) weight vectors extracted from a trained VQTAM (Vector-Quantized Temporal Associative Memory) model.

This subset contains the weight vectors of the current K first winning neurons.

The SOM-based TSP models have shown, among other things, that the SOM is a highly flexible neural architecture, in the sense that it can be successfully applied to unsupervised learning tasks (e.g. data clustering and visualization), as well as to typical supervised learning tasks (e.g. function approximation).

As regards to these algorithms proposed in different contexts, but taking again the majority for the same concepts, taxonomy was proposed to classify the unsupervised connexionnists networks developed for the space-time context. In an independent way, it was proposed to unify these various algorithms through a more general context, the General SOM for Structured Data (**GSOMSD**), definite for the treatment of any acyclic graph, as in the case of the temporal sequences.

Comparative table of features models:

Models:	Choice criterion BMU:	Weight updates and/or characteristics memory:
SOM	B= arg min $E_{i\,;\,i=1,\dots,N}$	$\Delta w_i = \gamma . h_{iv}(x(t) - w_i)$ For static data
TKM	B= arg max $V_i(t)$	$V_i(t) = d.V_i(t-1) - (\frac{1}{2}).\left\|x(t) - w_i(t)\right\|^2$ It is a SOM with internal STM mechanism, using leaky integrators
RSOM	B= arg min $Y_i(t)$	$y_i(t) = (1-\alpha).y_i(t-1) + \alpha.(x(t) - w_i(t))$ if α =1, RSOM=SOM with leaky integrators and internal STM

RecSOM	B= arg min $\{E_i\}$	$\Delta w_i^x = \gamma.h_{iv}.(x(t) - w_i^x)$ $\Delta w_i^y = \gamma.h_{iv}.(y(t-1) - w_i^y)$
CSOM	Like RecSOM	Like RecSOM
TSOM	B= arg min $\{a_i(t)\}$	$a_i(t) = (1-\alpha).\ a_i(t-1) + \alpha\ .e_i(t)^2$
Kangas	B= arg min $Y_i(t)$	$Y_i(t) = \Vert X(t) - W_{b(x)} \Vert \,/\, \Vert X(t) - W_i \Vert$
SARDNET	B= arg min $\{a_i(t)\}$	$a_i(t+1) = \alpha.\ a_i(t)$ with exponential decay ; internal STM

7. The GASOM paradigm

The hybridization of genetic algorithm GA with SOM abbreviates in GASOM has been proved to be well suited for finding global optima in complex search spaces. Its evaluation on well known problems shows that it effectively prevents premature convergence and seeks the global optimum. Its main features are respectively:

- An explicit representation of the search history coded in binary or real chromosomes.
- A random choice of population and initialization.
- A fitness evaluation promoting the best solution.
- Operator’s mechanism such as: selection, recombination, and mutation.

The fitness function applied on chromosomes involves population ordering with respect to the objective value (fitness). The evolution of the map SOM or recursive SOM, RecSOM, by genetic algorithm can imply vectors weight, topology, size (growth), connexion, self-referent rank for RecSOM, and iteration count. The implied element constitutes the individuals of the population or chromosomes. Genetic Algorithm are based on a set of this genomes, where each one represents a possible solution of the problem to be solved. The set is called a population. In an iterative process the chromosomes are adapted to the nature which is coded in the fitness-function. This induces an order on the population. Starting with an initial population we carry out in each iteration the selection, the recombination (crossover) and mutation steps. The chromosomes contained in the population before applying the recombination step are called the parent generation. Since the recombination step is used to create new chromosomes, we need an additional rule to keep the size of the population constant. The used rule is called survival of the fittest [18], [62], [70]. This rule states that only a constant number of the best chromosomes are used to form the next parent generation of the population. The minimum number of iterations required by the GA for a good solution depends on the implementation of the recombination and mutation step, the initial population and how the problem is coded in the chromosomes (binary or real codes). So that, each chromosome consists of a vector containing the code in temporal order.

Carla S. Moller-Levet and Hujun Yin in 'Circular SOM for temporal characterisation of modelled gene expressions', School of Electrical and Electronic Engineering, The University of Manchester, has remarked that one potential problem with the recurrent models is stability. In the case of temporal gene expression clustering, the data items presented to the map are not a spatial vector, but a sequence with time order in itself. They are time-series corresponding to the expression levels over time of a particular gene. Therefore, if a common 2-D SOM is used, the trained map can then be used to mark the trajectories of the expressions of the genes for comparison. In their approach, a self-organising latent lattice is proposed to identify pairs of genes with similar co-expressions. The co-expression of a pair of genes x and y is represented by a vector whose elements are the difference of present expression levels x(t) and y(t), and differences of slopes at present t, past (t−1) and future (t+1). All possible combinations of genes and their resulting sequence of vectors are selected to find out meaningful pairs of correlated genes. The vectors are mapped to the lattice and the patterns of trajectories of different pairs are compared. Similar trajectory patterns on the lattice may imply similar co-regulation patterns. There can be as many as millions possible pairs to examine and only pair-wise relations rather than group relations or clusters are revealed.

We approach the temporal extension of the SOM from another aspect, for example, the similarity metric. If the similarity metric considers temporal properties, then the neurons in the resultant map will exhibit temporal relationships. As time is one dimension, 1-D SOM is more appropriate. In addition, a circular, closed 1-D SOM, can further detect cyclic temporal characteristics [53].

In order to use the SOM based on the co-expression coefficient, the dot-product SOM is adopted and trained with norm one normalised data. When the dot -product is defined as the similarity metric between the input data and the weights, the learning equations should be modified accordingly by selecting the maximum of the dot-product as the winner node and by normalising the new weights at each step.

8. Conclusion

The SOM is a model of static data representation which constitutes a very powerful tool of classification and recognition. To justify it biologically, it should be instigated by dependence of the temporal context. One of the first temporal attempts at extension described in section 2, was the model of TKM founded by [Chappell and Taylor, 1993] substituting the activities of the SOM outputs by the leaky integrators outputs in the shapes of low frequency linear filters which integrate the activities of the passed inputs in the original algorithm of kohonen in a step to reach the real biological state of a neuron unit.

Section 3, relates to the model of the TKM modified while introducing the outputs of the 'leaky intégrators' at the inputs of the map to get a certain coherence with the standard algorithm of kohonen and to solve the problem of limited representations possibility in TKM; it was thus the recurrent SOM proposed by [Varsta, 1997].

Eventually, we will present at the level of the last section the recent work of [Thomas Voegtlin, 2002] introducing a self referent training class to algorithm SOM to develop the recursive SOM providing very encouraging results nowadays specially if it was hybridized with GA . In addition to this we aimed at closuring with other some recent temporal extensions of SOM under a comparative aspect.

In that, we deduce that all the algorithms briefly described above, modify algorithm SOM either by integrating the information of the passed iterations in those in progress, or by adding a context to the weights of the prototypes, in a form or another. Other work uses algorithm SOM as it is, in order to create classes of régresseurs, or possibly by using a local model of prediction inside each cluster, the model of linear prediction being or not linear. In the same way, it is possible to supplement the recursive approaches with local models, such as RecSOM with linear local models or local models of types k-more close neighbours. Moreover, it is possible to modify the vectors in entry of the SOM so that they take into account the awaited output of the model which is used to carry out a direct prediction as in Temporal Vector-Quantized Associative Memory (VQTAM).

Références

[1]. Thomas voegtlin : "Recursive Principal Components Analysis", Inria-campus scientifique, Nancy-France.
[2]. Marc strickert, Barbara hammer: "unsupervised recursive sequence processing" université de osnabrück-Germany, avril 2003.
[3]. Timo koskela, Markus varsta et Kaski: "Recurrent SOM with local linear models in time series prediction", Helsinki university of technologie-labo of computational engineering-Finland, avril 1998.
[4]. Romuald bone, Hubert cardot: "Apprentissage des delais dans les réseaux de neurones récurrents: application à la prevision de serie temporelles", université François-Rabelais de Tours, labo d'informatique, 2003.
[5]. Timo koskela, Markus varsta et Kaski : "Temporal sequence processing using recurrent SOM", Helsinki university of technologie-labo of computational engineering-Finland, avril 1998.
[6]. Thomas voegtlin 'thèse': "Réseaux de neurones et autoréférence", université lumière- lyon II, 2002.
[7]. Daniel L.James and Risto miikulainen: "Sardnet: a self organising feature map for sequences", Département of computer sciences, univ. of Texas at Austin, 1995.
[8]. Jean Marc Blanc: "Traitement de la prosodie par un réseau récurrent temporal", Institut des sciences cognitives, univ. Lumière lyon II, 6 avril 2004.
[9]. Geoffroy Simon : "Méthodes non linéaires pour séries temporelles", univ. Catholique de Louvain, labo de microélectronique, juin 2007.
[10]. S. Durand and F. Alexander: "TOM, A New Temporal Neural Net Architecture for Speech Signal Processing", in Proceedings of ICASSP '96.
[11]. B. Hammer, A. Micheli, A. Sperduti, and M. Strickert:" Recursive self-organizing network models", Neural Networks-Technical University of Clausthal to appear, 2004
[12]. M. Strickert and B. Hammer:" Neural Gas for Sequences", In T. Yamakawa, editor, Proceedings of the Workshop on Self-Organizing Networks (WSOM), pages 53–58, Kyushu Institute of Technology, 2003.
[13]. M. Hagenbuchner, A. Sperduti, and A. Tsoi: "A self-organizing map for adaptive processing of structured data", IEEE Transactions on Neural Networks, 2003.
[14]. Marc Strickert, Barbara Hammer:"Merge SOM for temporal data", Institute of Plant Genetics and Crop Plant Research Gatersleben and Technical University of Clausthal Preprint submitted to Elsevier Science 19 October 2004.
[15]. Stephan Bloehdorn and Sebastian Blohm: "A Self Organizing Map for Relation Extraction from
Wikipedia using Structured Data Representations", Institute AIFB, University of Karlsruhe
D-76128 Karlsruhe, Germany, 2006.
[16]. J. W. Miller and P. H. Lommel: "Biomimetic sensory abstraction using hierarchical quilted self-organizing maps", The Charles Stark Draper Laboratory, Inc.
555 Technology Square, Cambridge, MA 02139-3563, USA, 2006
[17]. Ozge Yeloglu, Student Member, IEEE, A. Nur Zincir-Heywood, Member, IEEE, Malcolm I. Heywood, Senior Member, IEEE: "Growing Recurrent Self Organizing Map", 2007.
[18]. Heni Ben Amor and Achim Rettinger: "Intelligent Exploration for Genetic Algorithms Using Self-Organizing Maps in Evolutionary Computation", February 4, 2005.
[19]. Kohonen, T.: "The self-organizing map", Proceedings of the IEEE, 78 (9), 1464-1480, 1990
[20]. Kohonen, T.: "Self-organizing maps", Springer, Berlin, 1995
[21]. Boucher, A., Seto, K.C., Journel, A.G.: "A novel method for mapping land cover changes: incorporating time and space with geostatistics", IEEE Transactions on Geoscience and Remote Sensing, 44 (11), 3427-3435, 2006

[22]. Dan Weaver and Nick Mount: "The use of Kohonen mapping for the elucidation of space-time trajectories of multiple parameters: potential applications in fluvial geomorphology", School of Geography, University of Nottingham, Nottingham NG7 2RD

[23]. M. Dittenbach, D. Merkl, A. Rauber, "The Growing Hierarchical Self-Organizing Map", Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, vol. 6, pp. 15-19, July 2000.

[24]. J. Zhou, Y. Fu, "Clustering High-Dimensional Data Using Growing SOM", Advances in Neural Networks ISNN 2005, Springer (2005), vol.3497, pp. 63-68.

[25]. M. Varsta, J. Heikkonnen, J. D. R. Millan, "Context Learning with the Self-Organizing Map", Proceedings of Workshop on Self-Organizing Maps, Helsinki University of Technology, pp. 197-202, 1997.

[26]. G. J. Chappell, J. G. Taylor, "The Temporal Kohonen Map", Neural Networks, vol. 6, pp. 441-445, 1995.

[27]. H.G. Kayacik, A.N. Zincir-Heywood, M.I. Heywood, "On the Capability of an SOM based Intrusion Detection System", IEEE-INNS

[28]. Hammer, B.: "Recurrent networks for structured data - a unifying approach and its properties", Cognitive Systems Research 3(2), 145-165, 2002

[29]. Hammer, B., Micheli, A., Sperduti, A., and Strickert, M.: "A general framework for unsupervised processing of structured data", Neurocomputing 57, 3-35, 2004

[30]. Kaski, S., Kangas, J., and Kohonen, T.: "Bibliography of self-organizing maps", papers: 1981-1997. Neural Computing Surveys. 1, 102-350. 1998

[31]. Martinetz, T., Berkovich, S., and Schulten, K.: "Neural-gas network for vector quantization and its application to time-series prediction". IEEE-Transactions on Neural Networks 4(4): 558-569, 1993

[32]. Neubauer, N.: "Recursive SOMs and Automata". M.Sc. Thesis, Cognitive Science, University of Osnabruck, 2005

[33]. Wiemer, J.C.: "The time-organized map algorithm: extending the self-organizing map to spatiotemporal signals", Neural Computation, 2003

[34]. M.Cottrell, S. Ibbou & P. Letrémy: "SOM-based algorithms for qualitative variables", *Neural Networks*, vol**,** 17, p.1149-1167, 2004

[35]. Markus Varsta Jukka Heikkonen Jouko Lampinen: "Analytical comparison of the Temporal Kohonen Map and the Recurrent Self Organizing Map", Laboratory of Computational Engineering Helsinki University of Technology-2000

[36]. M. Hagenbuchner, A. C. Tsoi, and A. Sperduti, "A supervised self organizing map for structured data," in Advances in Self-Organizing Maps, 2001, pp. 21–28.

[37]. T. Villmann1, U. Seiffert2, A.Wismüller3: "Theory and Applications of Neural Maps", 1Universität Leipzig, Klinik für Psychotherapie, Karl-Tauchnitz-Str. 25, 04107 Leipzig, Germany
2Leibniz Inst. of Plant Genetics and Crop Plant Research, Corrensstr. 3, 06466 Gatersleben, Germany
3LMU München, Institut für Klinische Radiologie, Klinikum Innenstadt, Ziemssenstr. 1, 80336 München, Germany

[38]. C. Aaron and Y. Tadjeddine: "Description of the group dynamic of funds managers using Kohonens map", In M. Verleysen, editor, Proc. Of Europ. Symp. On Artificial Neural Networks (ESANN'2004), in this book, Brussels, Belgium, 2004

[39]. Irini Reljin, Branimir Reljin, Gordana Jovanović: "Clustering and Mapping Spatial-Temporal Datasets Using SOM Neural Networks', JOURNAL OF AUTOMATIC CONTROL, UNIVERSITY OF BELGRADE; VOL. 13(1):55-60, 2003

[40]. B. Dorizzi, J.M.Duval and H. Debar : 'Utilisation de réseau récurrent pour la prévision de la consommation électrique', In Proc. Neural Networks and their Applications. Neuro Nimes France 1992.

[41]. Guilherme de a Baretto: 'Time in self organizing maps: an overview of models', International Journal of Computer Research, 10(2):139–179, Brazil 2001

[42]. V. Baier: "Motion perception with recurrent self-organizing maps based models", In Proceedings of the 2005 IEEE International Joint Conference on Neural Networks (IJCNN'05), pages 1182–1186, 2005.

[43]. Markus Varsta: "SELF-ORGANIZING MAPS IN SEQUENCE PROCESSING", Helsinki University of Technology Laboratory of Computational Engineering Publications Teknillisen korkeakoulun Laskennallisen tekniikan laboratorion julkaisuja Espoo 2002

[44]. Guilherme A. Barreto: "Time Series Prediction with the Self-Organizing Map: A Review", Department of Teleinformatics Engineering, Federal University of Cear´a Av. Mister Hull, S/N - C.P. 6005, CEP 60455-760, Center of Technology Campus of Pici, Fortaleza, Cear´a, Brazil, guilherme@deti.ufc.br

[45]. Varsta, M., Millán, J. Del R., and Heikkonen, J.: "A Recurrent Self Organizing Map for Temporal Sequence Processing", In *ICANN'97: International Conference on Artificial Neural Networks,* LNCS vol. 1327, pages 421–426. Springer, 1997

[46]. Kaski, S., Honkela, T., Lagus, K., and Kohonen, T.:"WEBSOM—self-organizing maps of document collections", Neurocomputing, 21(1):101–117, 1998

[47]. Koskela, T., Varsta, M., Heikkonen, J., and Kaski, K. (1998a): "Prediction using rsom with local linear models", Int. *Journal of Knowledge-Based Intelligent Engineering Systems*, 2(1):60–68.
[48]. Laaksonen, J., Koskela, M., Laakso, S., and Oja, E.: "PicSOM—content based image retrieval with self-organizing maps", Pattern *Recognition Letters*, 21(13–14):1199–1207, 2000
[49]. Lampinen, J. and Kostiainen, T.: "*Self Organizing Neural Networks: Recent Advances and applications"*, chapter Generative Probability Density Model in the SOM, pp.75–94. Physica-Verlag, 2001
[50]. Guilherme de a. barreto and Aluizio f. 'A taxonomy for spatiotemporal connectionist networks revisited: the unsupervised case', university of Paulo, Brazil-2003
[51]. G.A. Barreto and L.G.M. Souza.: "Adaptive filtering with the self-organizing map: a performance comparison", Neural Networks, 19(6-7):785–798, 2006.
[52]. Nikolaos Gianniotis, Peter Tino: "Visualisation of Tree-structured Data through Generative Probabilistic Modelling", School of Computer Science - University of Birmingham, Birmingham B15 2TT - UK
[53]. Carla S. Moller-Levet and Hujun Yin: "Circular SOM for temporal characterisation of Modelled gene expressions", School of Electrical and Electronic Engineering, The University of Manchester, Manchester M60 IQD, U.K.
[54] M. Cottrell, J. C. Fort, and G. Pages. : "Theoretical aspects of the SOM algorithm". Neurocomputing, 1998.
[55]. P. Demartines.: "Data Analysis through Self-Organized Neural Networks". PhD thesis, Grenoble University, Grenoble, France, 1995. (In french).
[56] J. A. Flanagan.: "Self-organization in the SOM and lebesque continuity of the input Distribution". In Proceedings of the International Joint Conference on Neural Networks, vol. 6, pages 26—31, Piscataway, NJ, 2000. Helsinki Univ of Technology, IEEE.
[57]. T. D. Hämäläinen.: "Parallel implementations of Self-Organizing Maps". In U. Seiffert and L. C. Jain, eds., Self-Organizing Neural Networks: Recent Advances and Applications, vol. 78 of Studies in Fuzziness and Soft Computing, chapter 11, pages 245—278. Springer-Verlag, Heidelberg, 2001.
[58]. B. Hammer, M. Strickert, and T. Villmann.: "Learning vector quantization for multimodal Data". In J. Dorronsoro, editor, Proc. International Conf. on Artificial Neural Networks (ICANN), LNCS 2415, pages 370—376. Springer Verlag, 2002.
[59]. B. Hammer, M. Strickert, and T. Villmann.: "On the generalization ability of GRLVQ Networks". P 249, Osnabrücker Zeitschriften zur Mathematik, University Osnabrück, Germany, 2003.
[60]. J. Lampinen and T. Kostiainen.: "Generative probability density model in the selforganizing Map". In U. Seiffert and L. Jain, eds., Self-Organizing Neural Networks, Studies in Fuzziness and Soft Computing, pages 75—92. Physica-Verlag, Heidelberg, New York, 2002.
[61]. M. Varsta, J. Heikkonen, J. Lampinen, and J. D. R. Millan.: "Temporal kohonen map and the recurrent self-organizing map: Analytical and experimental comparison". Neural Processing Letters, 13(3):237—251, June 2001.
[62] T. Villmann, B. Villmann, and V. Slowik.: "Evolutionary algorithms with neighbourhood cooperativness according neural maps". Neurocomputing, page in press, 2004.
[63]. Nikolaos Gianniotis, Peter Tino: "Visualisation of Tree-structured Data through Generative Probabilistic Modelling", School of Computer Science - University of Birmingham
Birmingham B15 2TT – UK ESANN'2007 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 25-27 April 2007, ISBN 2-930307-07-2.
[64]. Catherine Aaron, Corinne Perraudin, and Joseph Rynkiewicz: "Adaptation de l'algorithme SOM à l'analyse de données temporelles et spatiales : Application à l'étude de l'évolution des performances européennes en matière d'emploi', SAMOS – MATISSE Université de Paris 190 rue de Tolbiac, 75013 Paris, France, PAPER SUBMITTED TO THE SPECIAL ISSUE OF IEEE TNN ON TEMPORAL CODING 1
[65]. Guilherme A. Barreto and Aluizio F. R. Araujo: "Identication and Control of Dynamical Systems Using the Self-Organizing Map", Université catholique de Louvain Faculté des Sciences Appliquées Département d'Electricité Laboratoire de Microélectronique, 2004
[66]. Geoffroy Simon : 'Méthodes non linéaires pour séries temporelles : prédiction par Double Quantification Vectorielle et sélection de délai en hautes dimensions', Thèse présentée en vue de l'obtention du grade de Docteur en Sciences Appliquées Juin 2007
[67]. N. R. Euliano and J. C. Principe.: "A spatio-temporal memory based on SOMs with activity diffusion". In S. Oja and E. Kaski, editors, Kohonen Maps, pages 253–266.
Elsevier, Amsterdam, 1999
[68]. Barbara Hammer, Alessio Micheli, Nicolas Neubauer, Alessandro Sperduti, and Marc Strickert.: "Self-organizing maps for time series". In Proceedings of WSOM 2005, pages 115–122, Paris, France, September 2005.
[69]. Hämäläinen, A.: "Self-Organizing Map and Reduced Kernel Density Estimation", PhD thesis, Jyväskylä University, Jyväskylä, Finland, 1995

[70]. Ryosuke Kubota: "REPRODUCTION STRATEGY BASED ON SELF-ORGANIZING MAP FOR GENETIC ALGORITHMS", International Journal of Innovative Computing, Information and Control Volume 1, Number 4, December 2005.

Authors

Salhi Mohamed Salah received the Master degree in electrical engineering (signal processing & automatic) from the National School of Engineer of Tunis (ENIT-Tunisia). Actually, He is a PhD candidate in laboratory 'LSTS' of the same university.
His doctoral study focused on the speech recognition using dynamic Self Organizing Maps SOM and Evolutionary algorithm / GA,
E-Mail: salhi_mohamedsalah@yahoo.fr/ msalah.salhi@Gmail

N. Arous received computer science engineering degree from the National School of Computer Science (ENSI-Tunisia), the MS degree in electrical engineering (signal processing) from the National School of Engineer of Tunis (ENIT-Tunisia), the Ph. D. degree in electrical engineering (signal processing) from ENIT.
She is currently an associate professor in the computer science department at the High Institute of Computer Science of Tunis (ISI-Tunisia). Her research interests include scheduling optimization, artificial intelligence, pattern recognition, speech recognition and evolutionary neural networks.
Her teaching areas are artificial intelligence, logic programming, neural networks, genetic algorithms and pattern recognition.
E-mail:Najet.Arous@enit.rnu.tn, NajetArous@yahoo.fr

Ellouze received a Ph.D. degree in 1977 from l'Institut National

Paul Sabatier University (Toulouse-France), and Electronic Engineer Diploma

ENSEEIHT in 1968 at the same University.
In 1978, Dr. Ellouze joined the Department of Electrical Engineering at the National School of Engineer of Tunis (ENIT–Tunisia), as assistant professor

electronic, signal processing and computer architecture. In 1990, he became Professor in signal processing; digital signal processing and stochastic process. He has also served as director of electrical department at ENIT from 1978 to 1983. General Manager and President of the Research Institute on Informatics and Telecommunication IRSIT from 1987-1990, and President of the Institut in 1990-1994. He is now Director of Signal Processing Research Laboratory LSTS at ENIT, and is in charge of Control and Signal Processing Master degree at ENIT.
Pr Ellouze is IEEE fellow since 1987, he directed multiple Masters and Thesis and published over 200 scientific papers both in journals and proceedings. He is chief editor of the scientific journal Annales Maghrébines de l'Ingénieur. His research interest include neural networks and fuzzy classification, pattern recognition, signal processing and image processing applied in biomedical, multimedia, and man machine communication.
E-mail:N.Ellouze@enit.rnu.tn