Recurrent Oscillatory Self-organizing Map: Learning and Entrainment to Multiple Periodicities

Mauri Kaipainen

University of Helsinki, Cognitive Science Program
PL 13, 00014 Helsingin yliopisto, Finland &
Indiana University, Cognitive Science Program
406 Lindley Hall Bloomington, Indiana 47405 USA
kaipaine@helsinki.fi

Pantelis Papadopoulos
Indiana University, Computer Science Department, Cognitive Science Program,
406 Lindley Hall Bloomington, Indiana 47405 USA,
pantelis@cs.indiana.edu

Pasi Karhu
University of Helsinki, Cognitive Science Program
PL 13, 00014 Helsingin yliopisto, Finland
pasi.karhu@helsinki.fi

Abstract

A recurrent oscillatory self-organizing map (ROSOM) is introduced. This is an architecture that assumes oscillatory states, assigned to all units, indicating their "readiness-to-fire" or "exhaustion", and constitutes an internal timing mechanism. The architecture feeds the vector of all these states back into the input layer, making the units recognize and adapt to not only input but also to the entire history of activations. The self-organized map, so equipped, is capable of detecting sequences that are consistent over time in the input flow. The timing mechanism translates these sequences into real-time periodicities to which the model can automatically entrain itself. The network is shown to distinguish between sequence endpoints that differ only with respect to their temporal context, and to entrain to the periodicity of simple data, provided that the initial wavelength is set close enough to one of the data's salient periodicities. Furthermore, the network has, in principle, the capability of carrying on activity in the absence of any input.

1. Introduction

Patterns of behavior and environment appear to repeat themselves in a more or less regular temporal fashion, and the survival of a biological being may depend on its ability to detect periodicities and adapt its behavior to them. For humans, this temporal aspect of cognition, or entrainment, seems so natural that the prerequisite skills that make this possible are neither recognized nor appreciated. We argue that entrainment is not just a matter of an oscillator system recognizing an unambiguous input pulse, phase-locking and adjusting its rate to that pulse, assuming an a priori defined pattern that repeats itself coherently (McAuley 1995, Large and Kolen 1994). Such periodicities can be taken for granted only in mechanistic models that are not intended to capture essential aspects of cognition in an open environment. Real-life periodicities are hidden in a flow of multiple simultaneous features that describe the momentary system state. We suggest that full understanding of entrainment requires a model that performs, in a single intertwined process, both the entrainment and the concomitant classification of the feature space in which periodicities are embedded.

1.1. SOM

Our suggestion for such a model is based on the self-organizing map (SOM) first introduced by Kohonen (1982), a non-supervised artificial neural network paradigm, and it is a further elaboration of an earlier model by Kaipainen (1996). The widely adopted SOM algorithm can be thought of as a suggestion of how nature keeps constant statistics of type multidimensional scaling. It finds feature combinations that remain constant during an extended, potentially continuous and open-ended exposure to data. This property makes it very attractive from the dynamic point of view. However, the SOM, in its original form, is plainly static. Firstly, it requires the input to be given in random order, thus eliminating all sequential relations. Secondly, it depends on a fixed learning schedule and is therefore not applicable for continuous learning in open environments. It has no way of learning or recognizing temporal contexts of patterns.

1.2. Previous models

There have been many attempts to translate the SOM algorithm into sequential and temporal domains. The crucial question is how to allow the SOM to recognize and adapt to input vectors together with their temporal contexts. We shall limit ourselves to merely mentioning those models which incorporate two or more SOMs stacked one above the other, such that the map above accepts the activity of another below as its input (Kangas 1991, Morasso 1991, Zandhuis 1992). This solution seems to invite an infinite chain of inner observers, or homunculi, each recognizing the states of the previous one; but who or what will collect the final state of such a chain?

External mechanisms have been suggested (Kangas 1991, Kaipainen, Toiviainen and Louhivuori 1995a and 1995b, Kohonen 1991) to include context information in the SOM input, while some (James and Miikkulainen 1995) try to manage without using any context at all, relying on a method which prevents the same unit from winning several times, thus forcing winner-activities to draw trajectories on the map. This method, used also by Kaipainen, Toiviainen and Louhivuori, allows the SOM to accept serially ordered input without getting jammed by the activations of just a few units, but it does not guarantee context-dependent recognition of patterns. Kaipainen and Chappell and Taylor (1993) attempt to achieve sequential properties by assuming units that remember a short context in the manner of leaky integrators. This addition is sufficient to allow the map to develop winners for sequences of features instead of just static features. However, this kind of memory is limited to extremely short contexts, scarcely greater than one. More informative representations of context seem to be needed to handle longer sequences.

Finally, various suggestions have been made of how to build temporal activity waves that influence the choice of the winner. Hoekstra and 1993) and Kaipainen add lateral connections between all map units, generating a further activation which is used as an additional factor in determining the winner unit. A recent group of temporal SOM models assumes chemical-like mechanisms of temporal activity diffusion (Ruwisch, Bode and Purwins 1993, Euliano and Principe 1997, Cunningham and Waxman 1994, Kargupta and Ray 1994).

More related to our enterprise, Kangas (1990) introduces a recurrent SOM, in which output (i.e. the responding activity of every unit) is fed back to all units throughout the input layer. However, this solution lacks the internal timing mechanism necessary for entrainment.

2. Model

Our model is an oscillatory recurrent variant of the SOM. Its unique architectural solution lies in feeding back to the input layer the vector which comprises the states of all the units. Each state is a value assigned to a unit which oscillates as the function of the time elapsed since the unit's last winner-activation. This state creates a temporal activation wave which increases the unit's readiness to fire at time intervals that are dynamically adjusted to match salient periodicities of the data. The data periodicity, in turn, emerges from the organization of the map itself.

2.1.Architecture

ROSOM has a one-dimensional input layer and a two-dimensional map lattice, as in the standard SOM. Each map unit j, ranging between 1 and the number of map units k, is connected to all input units i with weights w_{ij}. These weights are assigned random values at time instant 0. Unit j is also characterized by response R_j to the total current input and age A_j which indicates the time since its last winner-activation. Each unit's readiness to win or its exhaustion is described by the unit's state S_j, a function of A_j, which will be defined later. At timestep n=0, $S_j(0) = 1$ for all j, reflecting the fact that, at that point, no unit should have an advantage over any other.

ROSOM is a recurrent variant of the SOM. All states are fed back as the internal input of the next time instant, encoding the total history of the network as the context of the input. The normalized external and internal inputs are concatenated to a total input: $U_i(n)=P'(n)$, $i \le I_{max}$, otherwise $U_i(n)=\gamma S'_{i-Imax}(n)$, where γ is a weighting parameter for the balance between internal vs. external input. The network accepts a sequence of patterns P_{mi} as external input, where i=1..I_{max}, m=1..l, m determines the pattern within the sequence of l patterns in total, and i determines the feature dimension (each pattern can be viewed as a feature vector of I_{max} dimensions). The pattern sequence P_{mi} cycles continuously, emulating periodicities of the environment. Generalization to an open input stream will not require major modifications.

2.2.Algorithm

The algorithm follows the basic process of the original SOM, in which the main steps are recognition (winner selection) and adaptation of weights. The additional time-related step could be called entrainment. The only major change to the original algorithm is the dynamic implementation of neighborhood control.

Recognition

At each time instant n, all map units respond to the total input U(n) based on how closely their weights match the input. Specifically, map unit j has response

(1) $$\mathbf{R}_j(n) = 1 - \mathbf{d}_j(n)$$, where

(2) $$\mathbf{d}_j(n) = \frac{1}{I_{\max} + k} \sum_{i=1}^{I_{\max}+k} \left(\left| \mathbf{W}_{ij}(n-1) - \mathbf{U}_i(n) \right|^R \right)^{\frac{Q}{R}}$$

is the normalized distance between the weights of map unit j and the current input U(n). In our applications, R=1 and

Q=1, implementing a city-block distance metric. The better the match, the closer the distance is to 0 and, consequently, the closer the response is to 1. The next step is to select the map unit *jmax* such that

$$\mathbf{R}_{j_{\max}}(n)\mathbf{S}^{\varepsilon}_{j_{\max}}(n-1) = \max_{j}\left\{\mathbf{R}_j(n)\mathbf{S}^{\varepsilon}_j(n-1)\right\} \quad (3)$$

and then call it the winner. In this competition, state S_j, an oscillatory function of A_j, has an effect whose strength is determined by drive parameter ε. S_j, can be interpreted as indicating the units' "readiness-to-fire" or "exhaustion" after recent firing. For the present purposes, we defined the function so as to describe an oscillatory behaviour around the value 1, gradually dampening as A_j increases:

(4)

$$\mathbf{S}_j(n) = \left(1 + a\frac{\dfrac{1}{1+\left|\sin(\pi A_j)\right|} - .75}{1+A_j}\right)\left(1 - \frac{1}{\left(1+A_j\right)^{Q}}\right)$$

where a defines the overall amplitude of the function and Q determines the decay profile. Figure 1. plots the function, showing maxima with a = 2 and Q = 4.

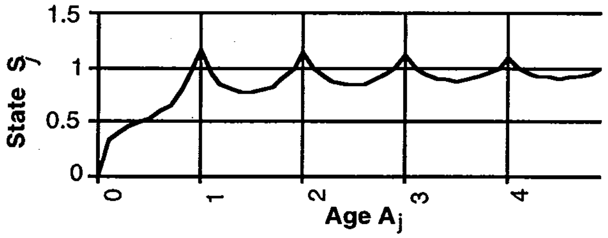

Fig. 1. State S_j as the function of A_j with a = 2 and Q = 4. Maxima are placed periodically, with the largest emphasis on the first, after which the amplitude of the waveform decreases gradually.

This function thus determines the oscillating advantage of each unit in the competition, emphasizing the first maximum and then neutralizing gradually. The period is adjusted dynamically, as explicated below.

Entrainment

Once the winner has been determined, the ages of all map units are adapted. The altered age of units is given by

$$\mathbf{A}_{j_{\max}}(n) = \tau\mathbf{A}_{\kappa},\; j = j_{\max} \quad (5)$$

$$\mathbf{A}_j(n) = \mathbf{A}_j(n-1) + T(n),\; j \neq j_{\max} \quad (6)$$

Thus the winner's age drops to a proportion of that age which corresponds to optimal readiness (determined by τ, a parameter with values between 0 and 1; usually 0). All other units become a bit "older", due to the increment of $T(n)$, the recovery factor. In the present implementation $T(n)$ is given by

$$T(n) = T(n-1) - \eta(1-e(n))\delta T(n) \quad (7)$$

where h is a constant that controls the speed of rate adaptation,

$$\delta T(n) = T(n-1)(\mathbf{A}_{j_{\max}}(n-1) - \mathbf{A}_{\kappa})\mathbf{R}_{j_{\max}}(n), \quad (8)$$

and error $e(n)$ is a leaky integration of how well the winners match inputs $e(n) = \Theta\, e(n-1) + (1-\Theta)\, d_{jmax}(n)$, where $d_{jmax}(n)$ is given by (2). We use Θ = .99 (slow change). This adaptation is moderated by error $e(n)$, assuming that it reflects the overall "goodness" of the map organization. Other implementations of the recovery factor control, based on different global measures of "goodness", are also conceivable. As can be seen by examining equation (8), $A(n)$ guides the adaptation of the recovery $T(n)$:

if $\mathbf{A}_{j_{\max}}(n-1) > \mathbf{A}_{\kappa}$ (winner aged too fast),

$\delta\mathbf{T}(n) > 0$, $\mathbf{T}(n) < \mathbf{T}(n-1)$ (slowing down), and

if $\mathbf{A}_{j}(n-1) < \mathbf{A}$ (winner aged too slowly),

$\delta\mathbf{T}(n) < 0$, so $\mathbf{T}(n) > \mathbf{T}(n-1)$ (speeding up).

Weight adaptation

The weights, corresponding to both input and context features, are adapted in the standard SOM manner. The winner, j_{max}, adapts its weights the most, and other units j adapt monotonically to a function Q of their actual distance $D(j, jmax)$ from the winner on the map.

$$\mathbf{W}_{ij}(n) = \mathbf{W}_{ij}(n-1) + \delta\mathbf{W}_{ij}(n) + N \quad (9)$$

Here N is simulated internal white noise in the interval $[-v,v]$, where $0<v<1$. $\delta W_{ij}(n)$ is given by

$$\delta\mathbf{W}_{ij}(n) = \left(\mathbf{W}_{ij}(n-1) - \mathbf{U}_i(n)\right)\beta Q(j, j_{\max}), \quad (10)$$

where $Q(j, j_{\max}) = e^{\left(\frac{D(j,j_{\max})}{\sigma(n)}\right)^2}$ and β is the learning rate. In our dynamic implementation of the neighborhood shrinking found in the original SOM, the exponential in the last equation is a Gaussian distribution centered at the winner, with standard deviation proportional to some function $\sigma(n)$ of $e(n)$. As the error decreases, $\sigma(n)$

decreases, which means that only a small neighbourhood around the current winner is adapted significantly.

3. Test data

In order to test the map on its ability to learn sequences with continuous values, we devised a method of tracking hand-drawn movements on a two-dimensional surface with a number of fixed reference points. A subject was asked to move the mouse near the fixed points labeled A, B, C, and D in the specified order ABACABAD (Fig. 2).The mouse coordinates were tracked at a sampling rate of 5 Hz, and for each tracked point i , the closeness g_{ij} to reference point j was computed by:

$g_{ij} = \xi / \left(h_{ij} + 1\right)$,where h_{ij} is the Euclidean distance between the tracked point j and reference point i, and ξ is a scaler to create a maximum value of 1. The data consisted of 23 four-dimensional vectors.

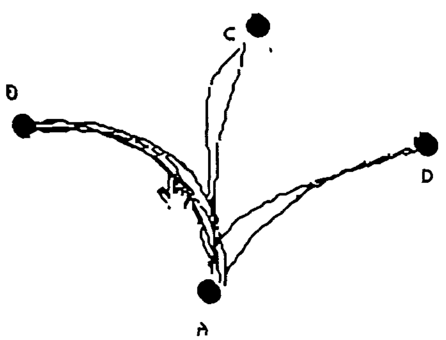

Fig. 2. Drawing sequence ABACABAD by freehand. Points are tracked, and for every point, a closeness feature is computed.

4. Results

We used three measures to monitor the performance of the network: the leaky average of error at the hub of the neighborhood (Err), the ratio of patterns with a uniquely selective unit (Rde), and the ratio of pattern responses that remain the same from previous to current epochs (Stb).

A network of 6 * 6 units was trained with the above data as input, with context weighting γ at 50% and learning rate β at 5% of error at the hub of the neighborhood. The simulation was quit manually when Stb remained at 100% for 10 subsequent epochs. The leaky average Err reached the .1 level after 55 epochs, and RDe and Stb settled at 100% after 78 epochs. Fig. 3 shows the established path of winner activations and the series of corresponding weight vectors after 78 epochs.

It can be observed that the path resembles the hand-drawn figure (Fig. 2) with four excursions from the A-region. The crucial difference in the data is that patterns sharing approximately the same coordinates on the data plotting, e.g. __2 and __6, are responded to by different units on the map, due to different contexts provided by the feedback from the units' states of the previous cycle. All units responsive to pattern A_, however, share the same region in the top right-hand corner, due to their roughly similar input. The left-hand graph displays the weights, with input features to the left and context to the right.

In order to probe the model's capabilities of rate adjustment and entrainment, we run a series of simulations, each with cycle durations ranging from 10 to 50 with steps of 2.5. All other parameters were same as in the previous experiment.

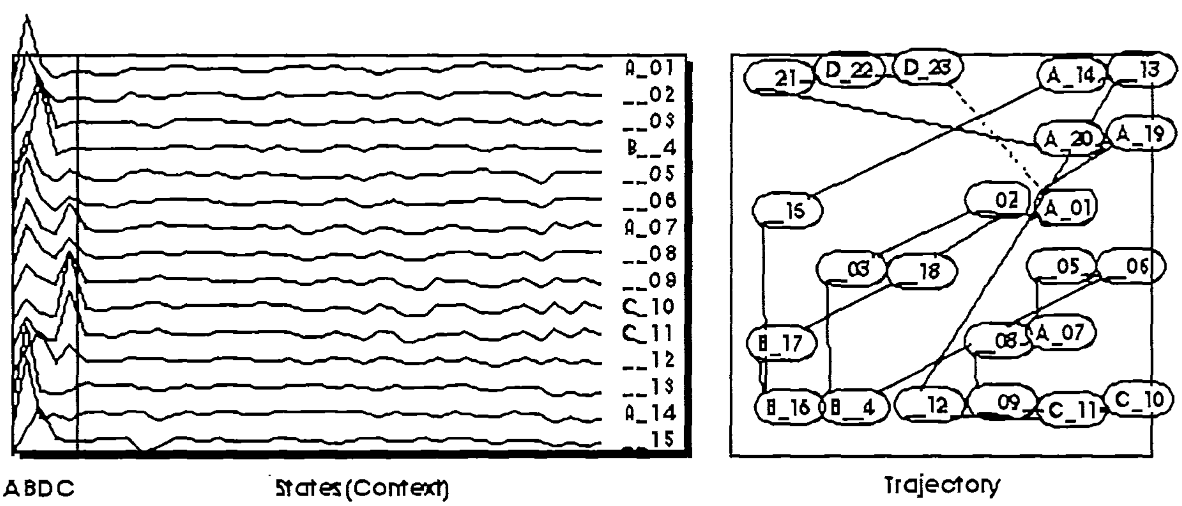

Fig. 3. Path of winner activations with pattern labels (right) and corresponding series of units' weight vectors (1 to 15), plotted as lines connecting hidden datapoints (left). Prominent peaks on the left correspond to closeness-features of the data.

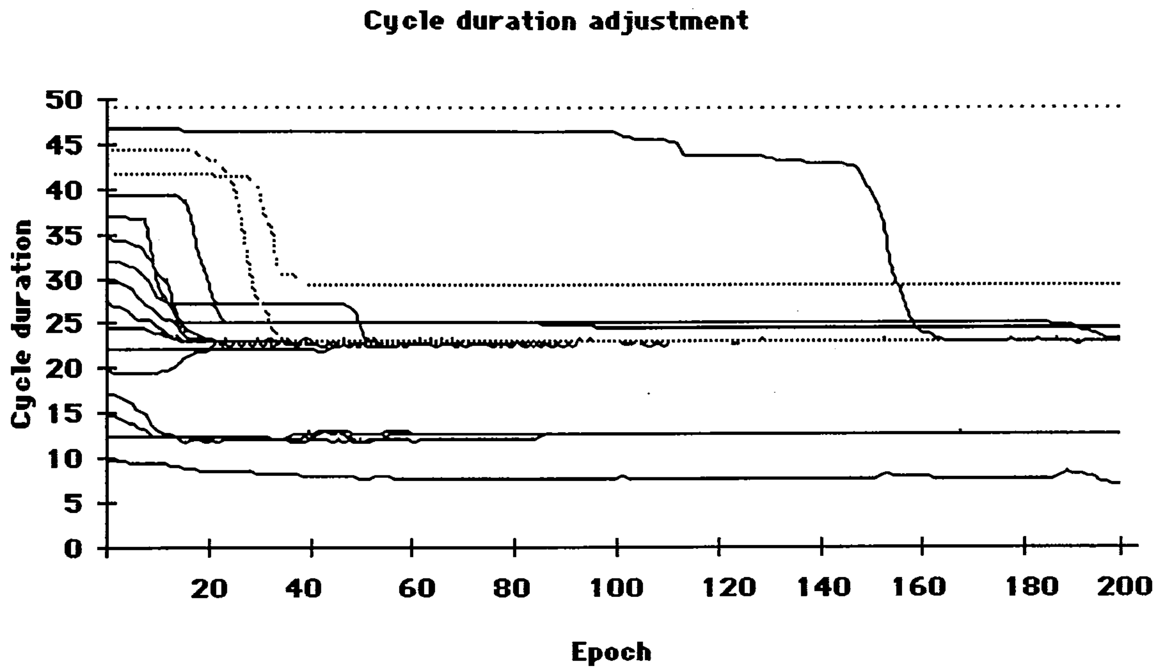

Fig. 4. Initial cycle durations converging to salient periodicities of the data (arrows at the right).

The network was able to detect periodicities within salient periodicities. The cycle duration was changed toward and eventually converged to attractor values which corresponded to the repetition structure of the data, the initial value determining the attractor. The strongest (10 simulations) attractor period was 23, which corresponded to the entire series ABACABAD. The second strongest attractor was 12 (3 simulations), which is near the half period, ABAX, X standing for "wildcard" accepting both C and D. There was also a simulation which appears to have been attracted by the 1/4 period AX, X accepting B, C, and D. It did not fully converge within the observed 200 epochs, however. Another simulation kept the cycle duration near 46 for a long time, but finally converged to 23 instead. Three simulations did not converge to any obvious periodicity within the observed simulation time.

5. Discussion

What is it that we actually recognize when we say that something repeats, recurs, or gives a rhythm or pulse? Numerous features of the body's environment run together from one instant to the next, while others coincide only occasionally and to a variable degree. To recognize a repeating pattern, a system has to detect consistencies over time and ignore merely occasional features. We have introduced a recurrent oscillatory self-organizing map, intended for such tasks. It is an architecture that assumes oscillatory readiness-to-fire states assigned to all units, and feedback of all these states back to the input layer.

The feedback loop makes the units both recognize and adapt to the entire history of activations on the map that goes with the multi-dimensional input flow. The self-organized map, so equipped, is capable of detecting patterns from the input flow which are consistent over time. These patterns constitute sequences which, in turn, are converted into real-time periodicities using the internal timing mechanism constituted by the units' states. The model entrains automatically to these periodicities by adjusting its global wavelength. With hand-drawn figure data, the network was shown to distinguish between sequence endpoints that differ only with respect to their temporal context. Further, it was shown to entrain to salient periodicities close enough to initial cycle duration values.

An important advantage of the internal timing mechanism is that the system will keep going even in the absence of input, i.e. the trained network has its own expectations of how the cycle should proceed. A demonstration of this property, however, falls outside the scope of this present paper.

Potential applications of the model include all oscillator-like perception and production tasks, which can be described as systems of multiple interconnected submechanisms, such as locomotion, or the production of repetitive movements needed for playing a musical instrument. At the current stage, the model is limited to quite obvious periodicities, like motor patterns of, say walking, or of musical rhythms. We realize that, in many domains, periodicities occur in more or less isolated regular patterns, such as the words of natural language. The adaptation of the model to such domains remains to be achieved, although it should not require major changes.

domains, periodicities occur in more or less isolated regular patterns, such as the words of natural language. The adaptation of the model to such domains remains to be achieved, although it should not require major changes.

In this experiment, the parameters were chosen by hand, the approach being that of trial and error. In the broader perspective, cognitive models should be thought of as being embedded in the dynamics of the entire body and environment, and the control parameters should emerge from such dynamics. Genetic algorithms are also considered as an optimization method for the further development of the model.

Acknowledgements

This study was carried out under an ASLA-Fulbright Research Grant to Mauri Kaipainen, and was supported by The Academy of Finland. Petri Toiviainen's help in various stages of the model is much appreciated.

References

Chappell, G.J.; Taylor, J. G. The Temporal Kohonen Map. *Neural Networks*, Vol. 6, Iss 3, 441-445, 1993.

Cunningham, R.K.; Waxman, A.M. Diffusion-Enhancement Bilayer: Realizing Long-Range Apparent Motion and Spatiotemporal Grouping in a Neural Architecture. *Neural Networks*, Vol 7, Nos. 6/7, pp. 895-924, 1994.

Euliano, N.R; Principe, J.C. Spatio-Temporal Self-Organizing Feature Maps. Computational Neuro-Engineering Laboratory, Department of Electrical Engineering, University of Florida. http://knicks.ee.ufl.edu/~neil/tempkoh/temporal.html, 1997.

Hoekstra, A.; Drossaers, M. F. An Extended Kohonen Feature Map for Sentence Recognition. -Gielen, S.; Kappen, B. (Eds.) Proceedings of the ICANN '93 (http://www.ph.tn.tudelft.nl/Research/neural/generalizatio/papers/icann93/icann93.html), 1993.

James, D.; Miikkulainen, R. (SARDNET: A Self-Organizing Feature Map for Sequences. *Advances in Neural Information Processing Systems*, 7., 1995.

Kaipainen, M. Representing and remotivating musical processes: modeling a recurrent musical ecology. Swets & Zeitlinger. *Journal of New Music Research*, Vol. 25, Iss. 2, 1996.

Kaipainen, M.; Toiviainen, P.; Louhivuori, J. A Self-organizing Map that Recognizes and Generates Melodies. -Pylkkänen, P.; Pylkkö, P. New Directions in Cognitive Science (Proceedings of the International Symposium, Saariselkä, 4-9 August). Helsinki: Finnish Artificial Intelligence Society. International Conferences, No 2, 1995.

Kaipainen, M.; Toiviainen, P.; Louhivuori, J. A Self-Organizing Map that Generates Melodies: Modeling and Testing Spontaneously Developing Implicit Grammars. Proceedings of the International Congress in Music & AI, University of Edinburgh, Computer Science, 1995.

Kangas, J. Representation of Time-Dependent Signals in Self-Organizing Maps. Unpublished licenciate's thesis: Helsinki University of Technology, Dept. of Computer Science, 1990.

Kangas, J. Time-dependent self-organizing maps for speech recognition. -Kohonen, T.; Mäkisara, K.; Simula, O.; Kangas, J. (Artificial Neural Networks. Proceedings of the ICANN-91, Espoo, Finland. North-Holland: Elsevier Science Publishers, 1991.

Kargupta, H.; Ray, S.R. Temporal Sequence Processing Based on the Biological Reaction-Diffusion Process. IEEE Proceedings of the International Conference on Neural Networks, Vol. 4, pp. 2315-2320, 1994.

Kohonen, T. The hypermap architecture. -Kohonen, T.; Mäkisara, K.; Simula, O.; Kangas, J. Artificial Neural Networks. Proceedings of the ICANN-91, Espoo, Finland. North-Holland: Elsevier Science, 1991.

Kohonen, T. Self-organized formation of topologically correct feature maps. *Biological Cybernetics* 43:59-69, 1982.

Large, E.; Kolen, J. F. Resonance and perception of musical meter. *Connection Science*, Vol. 6., Number 2 & 3., 1994.

McAuley, J.D. Perception of Time as Phase: Toward an Adaptive-Oscillator Model of Rhythm. PhD thesis, Indiana University, Bloomington, IN. Indiana University, Cognitive Science Program, Research Report 151, 1995.

Morasso, P. Self-organizing feature-maps for cursive script recognition. -Kohonen, T.; Mäkisara, K.; Simula, O.; Kangas, J. (Eds.) Artificial Neural Networks. Proceedings of the ICANN-91, Espoo, Finland. North-Holland: Elsevier Science Publishers, 1991.

Ruwisch, D.; Bode, M.; Purwins, H. G. Parallel hardware implementation of Kohonen's algorithm with an active. *Neural Networks*, Vol. 6, no. 8, pp. 1147-1157, 1993.

Zandhuis, J.A. Storing sequential data in self-organizing feature maps. Nijmegen, the Netherlands: Max-Planck-Insitute für Psycholinguistik. Internal Report MPI-NL-TG-4/92, 1992.