Theoretical Aspects of the SOM Algorithm

M.Cottrell[†], J.C.Fort[‡], G.Pagès[*]

† SAMOS/Université Paris 1

90, rue de Tolbiac, F-75634 Paris Cedex 13, France

Tel/Fax : 33-1-40-77-19-22, E-mail: cottrell@univ-paris1.fr

‡ Institut Elie Cartan/Université Nancy 1 et SAMOS

F-54506 Vandœuvre-Lès-Nancy Cedex, France

E-mail: fortjc@iecn.u-nancy.fr

* Université Paris 12 et Laboratoire de Probabilités /Paris 6

F-75252 Paris Cedex 05, France

E-mail:gpa@ccr.jussieu.fr

Abstract

The SOM algorithm is very astonishing. On the one hand, it is very simple to write down and to simulate, its practical properties are clear and easy to observe. But, on the other hand, its theoretical properties still remain without proof in the general case, despite the great efforts of several authors. In this paper, we pass in review the last results and provide some conjectures for the future work.

Keywords: Self-organization, Kohonen algorithm, Convergence of stochastic processes, Vectorial quantization.

1 Introduction

The now very popular SOM algorithm was originally devised by Teuvo Kohonen in 1982 [35] and [36]. It was presented as a model of the self-organization of neural connections. What immediatly raised the interest of the scientific community (neurophysiologists, computer scientists, mathematicians, physicists) was the ability of such a simple algorithm to produce organization, starting from possibly total disorder. That is called the *self-organization* property.

As a matter of fact, the algorithm can be considered as a generalization of the Competitive Learning, that is a Vectorial Quantization Algorithm [42], without any notion of neighborhood between the units.

In the SOM algorithm, a neighborhood structure is defined for the units and is respected throughout the learning process, which imposes the conservation of the neighborhood relations. So the weights are progressively updated according to the presentation of the inputs, in such a way that neighboring inputs are little by little mapped onto the same unit or neighboring units.

There are two phases. As well in the practical applications as in the theoretical studies, one can observe self-organization first (with large neighborhood and large adaptation parameter), and later on convergence of the weights in order to quantify the input space. In this second phase, the adaptation parameter is decreased to 0, and the neighborhood is small or indeed reduced to one unit (the organization is supposed not to be deleted by the process in this phase, that is really true for the 0-neighbor setting).

Even if the properties of the SOM algorithm can be easily reproduced by simulations, and despite all the efforts, the Kohonen algorithm is surprisingly resistant to a complete mathematical study. As far as we know, the only case where a complete analysis has been achieved is the *one dimensional case* (the input space has dimension 1) for a *linear network* (the units are disposed along a one-dimensional array).

A sketch of the proof was provided in the Kohonen's original papers [35], [36] in 1982 and in his books [37], [40] in 1984 and 1995. The first complete proof of both self-organization and convergence properties was established (for uniform distribution of the inputs and a simple step-neighborhood function) by Cottrell and Fort in 1987, [9].

Then, these results were generalized to a wide class of input distributions by Bouton and Pagès in 1993 and 1994, [6], [7] and to a more general neighborhood by Erwin et al. (1992) who have sketched the extension of the proof of self-organization [21] and studied the role of the neighborhood function [20]. Recently, Sadeghi [59], [60] has studied the self-organization for a general type of stimuli distribution and neighborhood function.

At last, Fort and Pagès in 1993, [26], 1995 [27], 1997 [3], [4] (with Benaim) have achieved the rigorous proof of the almost sure convergence towards a unique state, after self-organization, for a very general class of neighborhood functions.

Before that, Ritter et al. in 1986 and 1988, [52], [53] have thrown some light on the stationary state in any dimension, but they study only the final phase *after the self-organization*, and do not prove the existence of this stationary state.

In multidimensional settings, it is not possible to define what could be a *well ordered configuration set* that would be stable for the algorithm and that could be an absorbing class. For example, the grid configurations that Lo et al. proposed in 1991 or 1993, [45], [46] are not stable as proved in [10]. Fort and Pagès in 1996, [28] show that there is no organized absorbing set, at least when the stimuli space is continuous. On the other hand, Erwin et al. in 1992 [21] have proved that it is *impossible to associate a global decreasing potential function* to the algorithm, as long as the probability distribution of the inputs is continuous. Recently, Fort and

Pagès in 1994, [26], in 1996 [27] and [28], Flanagan in 1994 and 1996 [22], [23] gave some results in higher dimension, but these remain incomplete.

In this paper, we try to present the state of the art. As a continuation of previous paper [13], we gather the more recent results that have been published in different journals that can be not easily get-a-able for the neural community.

We do not speak about the variants of the algorithm that have been defined and studied by many authors, in order to improve the performances or to facilitate the mathematical analysis, see for example [5], [47], [58], [61]. We do not either address the numerous applications of the SOM algorithm. See for example the Kohonen's book [40] to have an idea of the profusion of these applications. We will only mention as a conclusion some original data analysis methods based on the SOM algorithm.

The paper is organized as follows: in section 2, we define the notations. The section 3 is devoted to the one dimensional case. Section 4 deals with the multidimensional 0-neighbor case, that is the simple competitive learning and gives some light on the quantization performances. In section 5, some partial results about the multidimensional setting are provided. Section 6 treats the discrete finite case and we present some data analysis methods derived from the SOM algorithm. The conclusion gives some hints about future researches.

2   Notations and definitions

The network includes n units located in an ordered lattice (generally in a one- or two-dimensional array). If $I = \{1, 2, \ldots, n\}$ is the set of the indices, the neighborhood structure is provided by a neighborhood function Λ defined on $I \times I$. It is symmetrical, non increasing, and depends only on the distance between i and j in the set of units I, (e.g. $\mid i - j \mid$ if $I = \{1, 2, \ldots, n\}$ is one-dimensional). $\Lambda(i, j)$ decreases with increasing distance between i and j, and $\Lambda(i, i)$ is usually equal to 1.

The input space Ω is a bounded convex subset of $\mathcal{R}^d$, endowed with the Euclidean distance. The inputs $x(t), t \geq 1$ are Ω-valued, independent with common distribution μ.

The network state at time t is given by

$$m(t) = (m_1(t),\ m_2(t),\ \ldots,\ m_n(t)).$$

where $m_i(t)$ is the d-dimensional weight vector of the unit i.

For a given state m and input x, the **winning** unit $i_c(x, m)$ is the unit whose weight $m_{i_c(x,m)}$ is the closest to the input x. Thus the network defines a map $\Phi_m : x \longmapsto i_c(x, m)$, from Ω to I, and the goal of the learning algorithm is to converge to a network state such the Φ_m map will be "topology preserving" in some sense.

For a given state m, let us denote $C_i(m)$ the set of the inputs such that i is the winning unit, that is $C_i(m) = \Phi_m^{-1}(i)$. The set of the classes $C_i(m)$ is the Euclidean Voronoï tessellation of the space Ω related to m.

The SOM algorithm is recursively defined by :

$$\begin{cases} i_c(x(t+1), m(t)) & = & \operatorname{argmin} \{\|x(t+1) - m_i(t)\|, i \in I\} \\ m_i(t+1) & = & m_i(t) - \varepsilon_t \Lambda(i_0, i)(m_i(t) - x(t+1)), \forall i \in I \end{cases} \quad (1)$$

The essential parameters are

- the dimension d of the input space

- the topology of the network

- the adaptation gain parameter ε_t, which is $]0, 1[$-valued, constant or decreasing with time,

- the neighborhood function Λ, which can be constant or time dependent,

- the probability distribution μ.

Mathematical available techniques

As mentioned before, when dealing with the SOM algorithm, one has to separate two kinds of results: those related to self-organization, and those related to convergence after organization. In any case, all the results have been obtained for a fixed time-invariant neighborhood function.

First, the network state at time t is a random Ω^n-valued vector $m(t)$ displaying as :

$$m(t+1) = m(t) - \varepsilon_t \, H(x(t+1), \, m(t)) \quad (2)$$

(where H is defined in an obvious way according to the updating equation) is a stochastic process. If ε_t and Λ are time-invariant, it is an homogeneous *Markov chain* and can be studied with the usual tools if possible (and fruitful). For example, if the algorithm converges in distribution, this limit distribution has to be an invariant measure for the Markov chain. If the algorithm has some fixed point, this point has to be an absorbing state of the chain. If it is possible to prove some strong organization [28], it has to be associated to an absorbing class.

Another way to investigate self-organization and convergence is to study the associated ODE (Ordinary Differential Equation) [41] that describes the mean behaviour of the algorithm :

$$\frac{dm}{dt} = - \, h(m) \quad (3)$$

where

$$h(m) = E(H(x, \, m)) = \int H(x, \, m) \, d\mu(x) \quad (4)$$

is the expectation of $H(., m)$ with respect to the probability measure μ.

Then it is clear that all the possible limit states $m^\star$ are solutions of the functional equation

$$h(m) = 0$$

and any knowledge about the possible attracting equilibrium points of the ODE can give some light about the self-organizing property and the convergence. But actually the complete asymptotic study of the ODE in the multidimensional setting seems to be untractable. One has to verify some *global assumptions* on the function h (and on its *gradient*) and the explicit calculations are quite difficult, and perhaps impossible.

In the *convergence phase*, the techniques depend on the kind of the desired convergence mode. For the *almost sure* convergence, the parameter ε_t needs to decrease to 0, and the form of equation (2) suggests to consider the SOM algorithm as a Robbins-Monro [57] algorithm.

The usual hypothesis on the adaptation parameter to get almost sure results is then:

$$\sum_t \varepsilon_t = +\infty \ \text{ and } \ \sum_t \varepsilon_t^2 < +\infty. \tag{5}$$

The less restrictive conditions $\sum_t \varepsilon_t = +\infty$ and $\varepsilon_t \searrow 0$ generally do not ensure the almost sure convergence, but some weaker convergence, for instance the convergence in probability.

Let us first examine the results in dimension 1.

3 The dimension 1

3.1 The self-organization

The input space is $[0, 1]$, the dimension d is 1 and the units are arranged on a linear array. The neighborhood function Λ is supposed to be *non increasing* as a function of the distance between units, the classical step neighborhood function satisfies this condition. The input distribution μ is *continuous* on $[0, 1]$: this means that it does not weight any point. This is satisfied for example by any distribution having a *density*.

Let us define

$$F_n^+ = \{m \in \mathcal{R} \, / \, 0 < m_1 < m_2 < \ldots < m_n < 1\}$$

and

$$F_n^- = \{m \in \mathcal{R} \, / \, 0 < m_n < m_{n-1} < \ldots < m_1 < 1\}.$$

In [9], [6], the following results are proved using Markovian methods :

Theorem 1 *(i) The two sets F_n^+ and F_n^- are absorbing sets.*
(ii) If ε is constant, and if Λ is decreasing as a function of the distance (e.g. if there are only two neigbors) the entering time τ, that is the hitting time of $F_n^+ \cup F_n^-$, is almost surely finite, and $\exists \lambda > 0$, s.t. $\sup_{m \in [0,1]^n} E_m(\exp(\lambda \tau))$ is finite, where E_m denote the expectation given $\mathbf{m}(0) = m$.

The theorem 1 ensures that the algorithm will almost surely order the weights. These results can be found for the more particular case (μ uniform and two neighbors) in Cottrell and Fort [9], 1987, and the succesive generalisations in Erwin et al. [21], 1992, Bouton and Pagès [6], 1993, Fort and Pagès [27], 1995, Flanagan [23], 1996.

The techniques are the Markov chain tools.

Actually following [6], it is possible to prove that whenever $\varepsilon \searrow 0$ and $\sum \varepsilon_t = +\infty$, then $\forall m \in [0,1]^n, \text{Proba}_m(\tau < +\infty) > 0$, (that is the probability of self-organization is positive regardless the initial values, but not a priori equal to 1). In [60], Sadeghi uses a generalized definition of the winner unit and shows that the probability of self-organization is uniformly positive, without assuming a lower bound for ε_t.

No result of almost sure reordering with a vanishing ε_t is known so far. In [10], Cottrell and Fort propose a still not proved conjecture: it seems that the re-organization occurs when the parameter ε_t has a $\frac{1}{\ln t}$ order.

3.2 The convergence for dimension 1

After having proved that the process enters an ordered state set (increasing or decreasing), with probability 1, it is possible to study the convergence of the process. So we assume that $\mathbf{m}(0) \in F_n^+$. It would be the same if $\mathbf{m}(0) \in F_n^-$.

3.2.1 Decreasing adaptation parameter

In [9] (for the uniform distribution), in [7], [27] and more recently in [3], [4], 1997, the almost sure convergence is proved in a very general setting. The results are gathered in the theorem below :

Theorem 2 *Assume that*
1) $(\varepsilon_t) \in]0,1[$ satisfies the condition (5),
2) the neighborhood function satisfies the condition H_Λ: there exists $k_0 < \frac{n-1}{2}$ such that $\Lambda(k_0 + 1) < \Lambda(k_0)$,
3) the input distribution μ satisfy the condition H_μ: it has a density f such that $f > 0$ on $]0,1[$ and $\ln(f)$ is strictly concave (or only concave, with $\lim_{0^+} f + \lim_{1^-} f$ positive),
Then
(i) The mean function h has a unique zero $m^\star$ in F_n^+.
(ii) The dynamical system $\frac{dm}{dt} = -h(m)$ is cooperative on F_n^+, i.e. the non diagonal elements of $\nabla h(m)$ are non positive.

(iii) $m^\star$ is attracting.
So if $\mathbf{m}(0) \in F_n^+$, $\mathbf{m}(t) \xrightarrow{a.s} m^\star$ almost surely.

In this part, the authors use the ODE method, a result by M.Hirsch on cooperative dynamical system [34], and the Kushner & Clark Theorem [41], [3]. A.Sadeghi put in light that the non-positivity of non-diagonal terms of ∇h is exactly the basic definition of a cooperative dynamical system and he obtained partial results in [59] and more general ones in [60].

We can see that the assumptions are very general. Most of the usual probability distributions (truncated on $[0, 1]$) have a density f such that $\ln(f)$ is strictly concave. On the other hand, the uniform distribution is not strictly ln-concave as well as the truncated exponential distribution, but both cumply the condition $\lim_{0+} f + \lim_{1-} f$ positive.

Condition (5) is essential, because if $\varepsilon_t \searrow 0$ and $\sum_t \varepsilon_t = +\infty$, there is only a priori convergence in probability.

In fact, by studying the associated ODE, Flanagan [22] shows that before ordering, it can appear metastable equilibria.

In the uniform case, it is possible to calculate the limit $m^\star$. Its coordinates are solutions of a $(n \times n)$-linear system which can be found in [37] or [9]. An explicit expression, up to the solution of a 3×3 linear system is proposed in [6]. Some further investigations are made in [31].

3.2.2   Constant adaptation parameter

Another point of view is to study the convergence of $\mathbf{m}(t)$ when $\varepsilon_t = \varepsilon$ is a constant. Some results are available when the neighborhood function corresponds to the two-neighbors setting. See [9], 1987, (for the uniform distribution) and [7], 1994, for the more general case. One part of the results also hold for a more general neighborhood function, see [3], [4].

Theorem 3 *Assume that $\mathbf{m}(0) \in F_n^+$,*
Part A: Assume that the hypotheses H_μ and H_Λ hold as in Theorem 2, then
For each $\varepsilon \in]0, 1[$, there exists some invariant probability ν^ε on F_n^+.
Part B: Assume only that $\Lambda(i, j) = 1$ if and only if $|i - j| = 0$ or 1 (classical 2-neighbors setting),
(i) If the input distribution μ has an absolutely continuous part (e.g. has a density), then for each $\varepsilon \in]0, 1[$, there exists a unique probability distribution ν^ε such that the distribution of $\mathbf{m}^t$ weakly converges to ν^ε when $t \longrightarrow \infty$. The rate of convergence is geometric. Actually the Markov chain is Doeblin recurrent.
(ii) Furthermore, if μ has a positive density, $\forall \varepsilon$, ν^ε is equivalent to the Lebesgue measure on F_n^+ if and only if n is congruent with 0 or 1 modulo 3. If n is congruent with 2 modulo 3, the Lebesgue measure is absolutely continuous with respect to ν^ε, but the inverse is not true, that is ν^ε has a singular part.

Part C: With the general hypotheses of Part A (which includes that of Part B), if $m^\star$ is the unique globally attractive equilibrium of the ODE (see Theorem 2), thus ν^ε converges to the Dirac distribution on $m^\star$ when $\varepsilon \searrow 0$.

So when ε is very small, the values will remain very close to $m^\star$.

Moreover, from this result we may conjecture that for a suitable choice of ε_t, certainly $\varepsilon_t = \frac{A}{\ln t}$, where A is a constant, both self-organization and convergence towards the unique $m^\star$ can be achieved. This could be proved by techniques very similar to the simulated annealing methods.

4 The 0 neighbor case in a multidimensional setting

In this case, we take any dimension d, the input space is $\Omega \subset \mathcal{R}^d$ and $\Lambda(i, j) = 1$ if $i = j$, and 0 elsewhere. There is no more topology on I, and *reordering* no makes sense. In this case the algorithm is essentially a stochastic version of the Linde, Gray and Buzo [44] algorithm (LBG). It belongs to the family of the vectorial quantization algorithms and is equivalent to the Competitive Learning. The mathematical results are more or less reachable. Even if this algorithm is deeply different from the usual Kohonen algorithm, it is however interesting to study it because it can be viewed as a limit situation when the neighborhood size decreases to 0.

The first result (which is classical for Competitive learning), and can be found in [54], [50], [39] is:

Theorem 4 *(i) The 0-neighbor algorithm derives from the potential*

$$V_n(m) = \frac{1}{2} \int \min_{1 \le i \le n} \|m_i - x\|^2 d\mu(x) \tag{6}$$

(ii) If the distribution probability μ is continuous (for example μ has a density f),

$$V_n(m) = \frac{1}{2} \sum_{i=1}^n \int_{C_i(m)} \|m_i - x\|^2 f(x) dx = \frac{1}{2} \int \min_{1 \le i \le n} \|m_i - x\|^2 f(x) dx \tag{7}$$

where $C_i(m)$ is the Voronoï set related with the unit i for the current state m.

The potential function $V_n(m)$ is nothing else than the *intra-classes variance* used by the statisticians to characterize the quality of a clustering. In the vectorial quantization setting, $V_n(m)$ is called *distortion*. It is a measure of the loss of information when replacing each input by the closest weight vector (or *code vector*). The potential $V_n(m)$ has been extensively studied since 50 years, as it can be seen in the Special Issue of IEEE Transactions on Information Theory (1982), [42].

The expression (7) holds as soon as $m_i \ne m_j$ for all $i \ne j$ and as the borders of the Voronoï classes have probability 0, $(\mu(\cup_{i=1}^n \partial C_i(m)) = 0)$. This last condition is

always verified when the distribution μ has a density f. With these two conditions, $V(m)$ is differentiable at m and its gradient vector reads

$$\nabla V_n(m) = \left(\int_{C_i(m)} (m_i - x) f(m) d(m) \right).$$

So it becomes clear ([50],[40]) that the Kohonen algorithm with 0 neighbor is the stochastic gradient descent relative to the function $V_n(m)$ and can be written :

$$m(t+1) = m(t) - \varepsilon_{t+1} \mathbf{1}_{C_i(m(t))}(x(t+1))(m(t) - x(t+1))$$

where $\mathbf{1}_{C_i(m(t))}(x(t+1))$ is equal to 1 if $x(t+1) \in C_i(m(t))$, and 0 if not.

The available results are more or less classical, and can be found in [44] and [8], for a general dimension d and a distribution μ satisfying the previous conditions.

Concerning the convergence results, we have the following when the dimension $d = 1$, see Pagès ([50], [51]), the Special Issue in IEEE [42] and also [43] for (ii):

The parameter $\varepsilon(t)$ has to satisfy the conditions (5).

Theorem 5 Quantization in dimension 1
(i) If ∇V_n has finitely many zeros in F_n^+, $m(t)$ converges almost surely to one of these local minima.
(ii) If the hypothesis H_μ holds (see Theorem (2)), V_n has only one zero point in F_n^+, say $m_n^\star$. This point $m_n^\star \in F_n^+$ and is a minimum. Furthermore if $m(0) \in F_n^+$, $m(t) \xrightarrow{a.s.} m_n^\star$.
(iii) If the stimuli are uniformly distributed on $[0, 1]$, then

$$m_n^\star = ((2i - 1)/2n)_{1 \leq i \leq n}.$$

The part (ii) shows that the global minimum de $V_n(m)$ is reachable in the one-dimensional case and the part *(iii)* is a confirmation of the fact that the algorithm provides an optimal discretization of continous distributions.

A weaker result holds in the d-dimensional case, because one has only the convergence to a *local minimum* of $V_n(m)$.

Theorem 6 Quantization in dimension d
If ∇V_n has finitely many zeros in F_n^+, and if these zeros have all their components pairwise distinct, $m(t)$ converges almost surely to one of these local minima.

In the d-dimensional case, we are not able to compute the limit, even in the uniform case. Following [48] and many experimental results, it seems that the minimum distortion could be reached for an hexagonal tesselation, as mentioned in [31] or [40].

In both cases, we can set the properties of the global minima of $V_n(m)$, in the general d-dimensional setting. Let us note first that $V_n(m)$ is invariant under any permutation of the integers $1, 2, \ldots, n$. So we can consider one of the global minima, the ordered one (for example the lexicographically ordered one).

Theorem 7 Quantization property
(i) The function $V_n(m)$ is continuous on $(\mathcal{R}^d)^n$ and reaches its (global) minima inside Ω^n.
(ii) For a fixed n, a point $m_n^\star$ at which the function V_n is minimum has pairwise distinct components.
(iii) Let n be a variable and $m_n^\star = (m_{n,1}^\star, m_{n,2}^\star, \ldots, m_{n,n}^\star)$ the ordered minimum of $V_n(m)$. The sequence $\min_{(\mathcal{R}^d)^n} V_n(m) = V_n(m_n^\star)$ converges to 0 as n goes to $+\infty$.
More precisely, there exists a speed $\beta = 2/d$ and a constante $A(f)$ such that

$$n^\beta V_n(m_n^\star) \longrightarrow A(f)$$

when n goes to $+\infty$.
Following Zador [64], the constant $A(f)$ can be computed, $A(f) = a_d \parallel f \parallel_\rho$, where a_d does not depend on f, $\rho = d/(d+2)$ and $\parallel f \parallel_\rho = [\int f^\rho(x)dx]^{1/\rho}$.
(iv) Then, the weighted empirical discrete probability measure

$$\mu_n = \sum_{i=1}^{n} \mu(C_i(m_n^\star))\delta_{m_{n,i}^\star}$$

converges in distribution to the probability measure μ, when $n \to \infty$.
(v) If F_n (resp. F) denotes the distribution function of μ_n (resp. μ), one has

$$\min_{(\mathcal{R}^d)^n} V_n(m) = \min_{(\mathcal{R}^d)^n} \int_\Omega (F_n(x) - F(x))^2 dx,$$

so when $n \to \infty$, F_n converges to F in quadratic norm.

The convergence in *(iv)* properly defines the *quantization property*, and explains how to reconstruct the input distribution from the n code vectors after convergence. But in fact this convergence holds for any sequence $y_n^\star = y_{1,n}, y_{2,n}, \ldots, y_{n,n}$, which "fills " the space when n goes to $+\infty$: for example it is sufficient that for any n, there exists an integer $n' > n$ such that in any interval $y_{i,n}, y_{i+1,n}$ (in $\mathcal{R}^d$), there are some points of $y_{n'}^\star$. But for any sequence of quantizers satisfying this condition, even if there is convergence in distribution, even if the speed of the convergence can be the same, the constant $A(f)$ will differ since it will not realize the minimum of the distortion.

For each integer n, the solution $m_n^\star$ which minimizes the quadratic distortion $V_n(m)$ and the quadratic norm $\parallel F_n - F \parallel^2$ is said to be *an optimal n-quantizer*. It ensures also that the discrete distribution function associated to the minimum $m_n^\star$ suitably weighted by the probability of the Voronoï classes, converges to the initial distribution function F. So the 0-neighbor algorithm provides a skeleton of the input distribution and as the distortion tends to 0 as well as the quadratic norm distance of F_n and F, it provides an *optimal quantizer*. The weighting of the Dirac functions by the volume of the Voronoï classes implies that the distribution μ_n is

usually quite different from the empirical one, in which each term would have the same weight $1/n$.

This result has been used by Pagès in [50] and [51] to numerically compute integrals. He shows that the speed of convergence of the approximate integrals is exactly $n^{\frac{2}{d}}$ for smooth enough functions, which is faster than the Monte Carlo method while $d \leq 4$.

The difficulty remains that the optimal quantizer $m_n^\star$ is not easily reachable, since the stochastic process $m(t)$ converges only to a local minimum of the distortion, when the dimension is greater than 1.

Magnification factor

There is *some confusion* [37], [52], between the asymptotic distribution of an *optimal quantizer $m_n^\star$* when $n \longrightarrow \infty$ and that one of the best *random quantizer*, as defined by Zador [64] in 1982.

The Zador's result, extended to the multi-dimensional case, is as follows : *Let f be the input density of the measure μ, and $(Y_1, Y_2, \ldots, Y_n)$ a random quantizer, where the code vectors Y_i are independent with common distribution of density g.*

Then, with some weak assumptions about f and g, the distortion tends to 0 when $n \longrightarrow \infty$, with speed $\beta = 2/d$, and it is possible to define the quantity

$$A(f,g) = \lim_{n \longrightarrow \infty} n^\beta E_g[\sum_{i=1}^n \int_{C_i} \|Y_i - x\|^2 f(x) dx]$$

Then for any given input density f, the density g (assuming some weak condition) which minimises $A(f,g)$ is

$$g^\star \sim C \; f^{d/d+2}.$$

The inverse of the exponent $d/(d+2)$ is refered as *Magnification Factor*. Note that in any case, when the data dimension is large, this exponent is near 1 (it value is 1/3 when $d = 1$). Note also that this power has no effect when the density f is uniform. But in fact the optimal quantizer is another thing, with another definition.

Namely the optimal quantizer $m_n^\star$ (formed with the code vectors $m_{1,n}^\star, m_{2,n}^\star, \ldots, m_{n,n}^\star$), minimizes the distortion $V_n(m)$, and is got after convergence of the 0-neighbor algorithm (if we could ensure the convergence to a global minimum, that is true only in the one-dimensional case). So if we set

$$A_n(f, m_n^\star) = n^\beta V_n(m_n^\star) = n^\beta \sum_{i=1}^n \int_{C_i} \|m_{i,n}^\star - x\|^2 f(x) dx$$

actually we have,

$$A(f) = \lim_{n \longrightarrow \infty} A_n(f, m_n^\star) < A(f, g^\star)$$

and the limit of the discrete distribution of $m_n^\star$ is not equal to $g^\star$. *So there is no magnification factor, for the 0-neighbor algorithm as claimed in many papers. It can be an approximation, but no more.*

The problem comes from the confusion between two distinct notions: random quantizer and optimal quantizer. And in fact, the good property is the convergence of the weighted distribution function (7).

As to the SOM algorithm in the one-dimensional case, with a neighborhood function not reduced to the 0-neighbor case, one can find in [55] or [19] some result about a possible limit of the discrete distribution when the number of units goes to ∞. But actually, the authors use the Zador's result which is not appropriate as we just see.

5　The multidimensional continuous setting

In this section, we consider a general neighborhood function and the SOM algorithm is defined as in Section 2.

5.1　Self-organization

When the dimension d is greater than 1, little is known on the classical Kohonen algorithm. The main reason seems to be the fact that it is difficult to define what can be an *organized state* and that no *absorbing* sets have been found. The configurations whose coordinates are monotoneous are not stable, contrary to the intuition. For each configuration set which have been claimed to be left stable by the Kohonen algorithm, it has been proved later that it was possible to go out with a positive probability. See for example [10]. Most people think that the Kohonen algorithm in dimension greater than 1 could correspond to an irreducible Markov chain, that is a chain for which there exists always a path with positive probability to go from anywhere to everywhere. That property imply that there is no absorbing set at all.

Actually, as soon as $d \geq 2$, for a constant parameter ε, the 0-neighbor algorithm is an Doeblin recurrent irreducible chain (see [7]), that cannot have any absorbing class.

Recently, two apparently contradictory results were established, that can be collected together as follows.

Theorem 8 *(d = 2 and ε is a constant) Let us consider a $n \times n$ units square network and the set F^{++} of states whose both coordinates are separately increasing as function of their indices, i.e.*

$$F^{++} = \left\{ \forall i_1 \leq n, m^2_{i_1,1} < m^2_{i_1,2} < \ldots < m^2_{i_1,n}, \forall i_2 \leq n, m^1_{1,i_2} < m^1_{2,i_2} < \ldots < m^1_{n,i_2} \right\}$$

(i) If μ has a density on Ω, and if the neighborhood function Λ is everywhere positive and decreases with the distance, the hitting time of F^{++} is finite with positive probability (i.e. > 0, but possibly less than 1). See Flanagan ([22], [23]).
(ii) In the 8-neighbor setting, the exit time from F^{++} is finite with positive probability. See Fort and Pagès in ([28]).

This means that (with a constant, even very small, parameter ε), the organization is temporarily reached and that even if we guess that it is almost stable, dis-organization may occur with positive probability.

More generally, the question is how to define an organized state. Many authors have proposed definitions and measures of the self-organization, [65], [18], [62], [32], [63], [33]. But none such "organized" sets have a chance to be absorbing.

In [28], the authors propose to consider that *a map is organized if and only if the Voronoï classes of the closest neighboring units are contacting*. They also precisely define the nature of the organization (strong or weak).

They propose the following definitions :

Definition 1 Strong organization
There is strong organization if there exists a set of organized states $\mathcal{S}$ such that
(i) $\mathcal{S}$ is an absorbing class of the Markov chain $m(t)$,
(ii) The entering time in $\mathcal{S}$ is almost surely finite, starting from any random weight vectors (see [6]).

Definition 2 Weak organization
There is weak organization if there exists a set of organized states $\mathcal{S}$ such that all the possible attracting equilibrium points of the ODE defined in 3 belong to the set $\mathcal{S}$.

The authors prove that there is no strong organization at least in two seminal cases: the input space is $[0,1]^2$, the network is one-dimensional with two neighbors or two-dimensional with eight neighbors. The existence of weak organization should be investigated as well, but until now no exact result is available even if the simulations show a stable organized limit behavior of the SOM algorithm.

5.2 Convergence

In [27], (see also [26]) the gradient of h is computed in the d-dimensional setting (when it exists). In [53], the convergence and the nature of the limit state is studied, assuming that the organization has occured, although there is no mathematical proof of the convergence.

Another interesting result received a mathematical proof thanks to the computation of the gradient of h: it is the dimension selection effect discovered by Ritter and Schulten (see [53]). The mathematical result is (see [27]:

Theorem 9 *Assume that $m_1^\star$ is a stable equilibrium point of a general d_1-dimensional Kohonen algorithm, with n_1 units, stimuli distribution μ_1 and some neighborhood function Λ. Let μ_2 be a d_2-dimensional distribution with mean $m_2^\star$ and covariance matrix Σ_2. Consider the $d_1 + d_2$ Kohonen algorithm with the same units and the same neighborhood function. The stimuli distribution is now $\mu_1 \otimes \mu_2$.*

Then there exists some $\eta > 0$, such that if $\|\Sigma_2\| < \eta$, the state $m_1^\star$ in the subspace $m_2 = m_2^\star$ is still a stable equilibrium point for the $d_1 + d_2$ algorithm.

It means that if the stimuli distribution is close to a d_1-dimensional distribution in the $d_1 + d_2$ space, the algorithm can find a d_1-space stable equilibrium point. That is the *dimension selection effect*.

From the computation of the gradient ∇h, some partial results on the stability of grid equilibriums can also be proved:

Let us consider $I = I_1 \times I_2 \times \ldots \times I_d$ a d-dimensional array, with $I_l = \{1, 2, \ldots, n_l\}$, for $1 \leq l \leq d$. Let us assume that the neighborhood function is a product function (for example 8 neighbors for $d = 2$) and that the input distributions in each coordinate are independent, that is $\mu = \mu_1 \otimes \ldots \otimes \mu_d$. At last suppose that the support of each μ_l is $[0,1]$.

Let us call *grid states* the states $m^\star = (m^\star_{i_l l}, 1 \leq i_l \leq n_l, 1 \leq l \leq d)$, such that for every $1 \leq l \leq d$, $(m^\star_{i_l l}, 1 \leq i_l \leq n_l)$ is an equilibrium for the one-dimensional algorithm. Then the following results hold [27] :

Theorem 10 *(i) The grid states are equilibrium points of the ODE (3) in the d-dimensional case.*
(ii) For $d = 2$, if μ_1 and μ_2 have strictly positive densities f_1 and f_2 on $[0, 1]$, if the neighborhood functions are strictly decreasing, the grid equilibrium points are not stable as soon as n_1 is large enough and the ratio $\frac{n_1}{n_2}$ is large (or small) enough (i.e. when $n_1 \longrightarrow +\infty$ and $\frac{n_1}{n_2} \longrightarrow +\infty$ or 0, see [27], Section 4.3).
(iii) For $d = 2$, if μ_1 and μ_2 have strictly positive densities f_1 and f_2 on $[0, 1]$, if the neighborhood functions are degenerated (0 neighbor case), $m^\star$ is stable if n_1 and n_2 are less or equal to 2, is not stable in any other case (may be excepted when $n_1 = n_2 = 3$).

The (ii) gives a negative property for the non square grid which can be related with this one: the product of one-dimensional quantizers is not the correct vectorial quantization. But also notice that we have no result about the simplest case: the square grid equilibrium in the uniformly distributed case. Everybody can observe by simulation that this square grid is stable (and probably the unique stable "organized" state). Nevertheless, even if we can numerically verify that it is stable, using the gradient formula it is not mathematically proved even with two neighbors in each dimension!

Moreover, if the distribution μ_1 and μ_2 are not uniform, generally the square grids are not stables, as it can be seen experimentally.

6 The discrete case

In this case, there is a finite number N of inputs and $\Omega = \{x_1, x_2, \ldots, x_N\}$. The input distribution is uniform on Ω that is $\mu(dx) = \frac{1}{N} \sum_{l=1}^{N} \delta_{x_l}$. It is the setting of many practical applications, like Classification or Data Analysis.

6.1 The results

The main result ([39], [56]) is that for not time-dependent general neighborhood, the algorithm locally derives from the potential

$$
\begin{aligned}
V_n(m) &= \frac{1}{2N} \sum_{i=1}^{n} \sum_{x_l \in C_i(m)} \left(\sum_{j=1}^{n} \Lambda(i-j) \|m_j - x_l\|^2 \right) \\
&= \frac{1}{2} \sum_{i=1}^{n} \int_{C_i(m)} \sum_{j=1}^{n} \Lambda(i-j) \|m_j - x\|^2 \mu(dx) \\
&= \frac{1}{2} \sum_{i,j=1}^{n} \Lambda(i-j) \int_{C_i(m)} \|m_j - x\|^2 \mu(dx).
\end{aligned}
$$

When $\Lambda(i,j) = 1$ if i and j are neighbors, and if $\mathcal{V}(j)$ denotes the neighborhood of unit i in I, $V_n(m)$ also reads

$$
V_n(m) = \frac{1}{2} \sum_{j=1}^{n} \int_{\cup_{i \in \mathcal{V}(j)} C_i(m)} \|m_j - x\|^2 \mu(dx).
$$

$V_n(m)$ is an *intra-class variance extended to the neighbor classes* which is a generalization of the distortion defined in Section 4 for the 0-neighbor setting. But this potential does have many singularities and its complete analysis is not achieved, even if the discrete algorithm can be viewed as a stochastic gradient descent procedure. In fact, there is a problem with the borders of the Voronoï classes. The set of all these borders along the process $\mathbf{m}(t)$ trajectories has measure 0, but it is difficult to assume that the given points x_l never belong to this set.

Actually the potential is the true measure of the self-organization. It measures both clustering quality and proximity between classes. Its study should provide some light on the Kohonen algorithm even in the continuous case.

When the stimuli distribution is continuous, we know that the algorithm is not a gradient descent [21]. However the algorithm can be seen then as an approximation of the stochastic gradient algorithm derived from the function $V_n(m)$. Namely, the gradient of $V_n(m)$ has a non singular part which corresponds to the Kohonen algorithm and a singular one which prevents the algorithm to be a gradient descent.

This remark is the base of many applications of the SOM algorithm as well in combinatorial optimization, data analysis, classification, analysis of the relations between qualitative classifying variables.

6.2 The applications

For example, in [24], Fort uses the SOM algorithm with a close one-dimensional string, in a two dimensional space where are located M cities. He gets very quickly a very good sub-optimal solution. See also the paper [1].

The applications in data analysis and classification are more classical. The principle is very simple: after convergence, the SOM algorithm provides a two(or one)-dimensional organized classification which permit a low dimensional representation of the data. See in [40] an impressive list of examples.

In [15] and [17], an application to forecasting is presented from a previous classification by a SOM algorithm.

6.3 Analysis of qualitative variables

Let us define here two original algorithms to analyse the relations between qualitative variables. The first one is defined only for two qualitative variables. It is called KORRESP and is analogous to the simple classical Correspondence Analysis. The second one is devoted to the analysis of any finite number of qualitative variables. It is called KACM and is similar to the Multiple Correspondence Analysis. See [11], [14], [16] for some applications.

For both algorithms, we consider a sample of individuals and a number K of questions. Each question $k, k = 1, 2, \ldots, K$ has m_k possible answers (or modalities). Each individual answers each question by choosing one and only one modality. If $M = \sum_{1 \leq k \leq} m_k$ is the total number of modalities, each individual is represented by a row M-vector with values in $0, 1$. There is only one 1 between the 1st component and the m_1-th one, only one 1 between the $m_1 + 1$-th component and the $m_1 + m_2$-th one and so on.

In the general case where $M > 2$, the data are summarized into a Burt Table which is a cross tabulation table. It is a $M \times M$ symmetric matrix and is composed of $K \times K$ blocks, such that the (k, l)-block B_{kl} (for $k \neq l$) is the $(m_k \times m_l)$ contingency table which crosses the question k and the question l. The block B_{kk} is a diagonal matrix, whose diagonal entries are the numbers of individuals who have respectively chosen the modalities $1, 2, \ldots, m_k$ for question k. In the following, the Burt Table is denoted by B.

In the case $M = 2$, we only need the contingency table T which crosses the two variables. In that case, we set p (resp. q) for m_1 (resp. m_2).

The KORRESP algorithm

In the contingency table T, the first qualitative variable has p levels and corresponds with the rows. The second one has q levels and corresponds with the columns. The entry n_{ij} is the number of individuals categorized by the row i and the column j. From the contingency table, the matrix of relative frequencies ($f_{ij} = n_{ij}/(\sum_{ij} n_{ij})$) is computed.

Then the rows and the columns are normalized in order to have a sum equal to 1. The row profile $r(i), 1 \leq i \leq p$ is the discrete probability distribution of the second variable given that the first variable has modality i and the column profile $c(j), 1 \leq j \leq q$ is the discrete probability distribution of the first variable given

that the second variable has modality j. The classical Correspondence Analysis is a simultaneous weighted Principal Component Analysis on the row profiles and on the column profiles. The distance is chosen to be the χ^2 distance. In the simultaneous representation, related modalities are projected into neighboring points.

To define the algorithm KORRESP, we build a new data matrix $\mathcal{D}$: to each row profile $r(i)$, we associate the column profile $c(j(i))$ which maximizes the probability of j given i, and conversely, we associate to each column profile $c(j)$ the row profile $r(i(j))$ the most probable given j. The data matrix $\mathcal{D}$ is the $((p+q) \times (q+p))$-matrix whose first p rows are the vectors $(r(i), c(j(i)))$ and last q rows are the vectors $(r(i(j)), c(j))$. The SOM algorithm is processed on the rows of this data matrix $\mathcal{D}$. Note that we use the χ^2 distance to look for the winning unit and that we alternatively pick at random the inputs among the p first rows and the q last ones. After convergence, each modality of both variables is classified into a Voronoï class. Related modalities are classified into the same class or into neighboring classes. This method give a very quick, efficient way to analyse the relations between two qualitative variables. See [11] and [12] for real-world applications.

The KACM Algorithm

When there are more than two qualitative variables, the above method does not work any more. In that case, the data matrix is just the Burt Table B. The rows are normalized, in order to have a sum equal to 1. At each step, we pick a normalized row at random according to the frequency of the corresponding modality. We define the winning unit according to the χ^2 distance and update the weights vectors as usual. After convergence, we get an organized classification of all the modalities, where related modalities belong to the same class or to neighboring classes. In that case also, the KACM method provides a very interesting alternative to classical Multiple Correspondence Analysis.

The main advantages of both KORRESP and KACM methods are their rapidity and their small computing time. While the classical methods have to use several representations with decreasing information in each, ours provide only one map, that is rough but unique and permit a rapid and complete interpretation. See [14] and [16] for the details and financial applications.

7   Conclusion

So far, the theoretical study in the one-dimensional case is nearly complete. It remains to find the convenient decreasing rate to ensure the ordering. For the multidimensional setting, the problem is difficult. It seems that the Markov chain is irreducible and that further results could come from the careful study of the Ordinary Differential Equation (ODE) and from the powerful existing results about the cooperative dynamical systems.

On the other hand, the applications are more and more numerous, especially in data analysis, where the representation capability of the organized data is very valuable. The related methods make up a large and useful set of methods which can be substituted to the classical ones. To increase their use in the statistical community, it would be necessary to continue the theoretical study, in order to provide quality criteria and performance indices with the same rigour as for the classical methods.

Acknowledgements

References

[1] B.Angéniol, G.de la Croix Vaubois, J.Y. Le Texier, Self-Organizing Feature Maps and the Travelling Salesman Problem, *Neural Networks*, Vol.1, 289-293, 1988.

[2] M.Benaïm, Dynamical System Approach to Stochastic Approximation, *SIAM J. of Optimization*, 34, 2, 437-472, 1996.

[3] M.Benaïm, J.C.Fort, G.Pagès, Almost sure convergence of the one-dimensional Kohonen algorithm, *Proc. ESANN'97*, M.Verleysen Ed., Editions D Facto, Bruxelles, 193-198, 1997.

[4] M.Benaïm, J.C.Fort, G.Pagès, Convergence of the one-dimensional Kohonen algorithm, submitted.

[5] C.M.Bishop, M.Svensn, C.K.I. Williams, GTM: the generative topographic mapping, to appear in *Neural Computation*, 1997.

[6] C.Bouton, G.Pagès, Self-organization of the one-dimensional Kohonen algorithm with non-uniformly distributed stimuli, *Stochastic Processes and their Applications*, 47, 249-274, 1993.

[7] C.Bouton, G.Pagès, Convergence in distribution of the one-dimensional Kohonen algorithm when the stimuli are not uniform, *Advanced in Applied Probability*, 26, 1, 80-103, 1994.

[8] C.Bouton, G.Pagès, About the multi-dimensional competitive learning vector quantization algorithm with a constant gain, *Annals of Applied Probability*, 7, 3, 670-710, 1997.

[9] M.Cottrell, J.C.Fort, Etude d'un algorithme d'auto-organisation, *Ann. Inst. Henri Poincaré*, 23, 1, 1-20, 1987.

[10] M.Cottrell, J.C.Fort, G.Pagès, Comments about Analysis of the Convergence Properties of Topology Preserving Neural Networks, *IEEE Transactions on Neural Networks*, Vol. 6, 3, 797-799, 1995.

[11] M.Cottrell, P.Letremy, E.Roy, Analysing a contingency table with Kohonen maps : a Factorial Correspondence Analysis, *Proc. IWANN'93*, J.Cabestany, J.Mary, A.Prieto Eds., Lecture Notes in Computer Science, Springer, 305-311, 1993.

[12] M.Cottrell, P.Letremy, Classification et analyse des correspondances au moyen de l'algorithme de Kohonen : application à l'étude de données socio-économiques, *Proc. Neuro-Nîmes*, 74-83, 1994.

[13] M.Cottrell, J.C.Fort, G.Pagès, Two or Three Things that we know about the Kohonen Algorithm, *Proc. ESANN'94*, M.Verleysen Ed., Editions D Facto, Bruxelles, 235-244, 1994.

[14] M.Cottrell, S.Ibbou, Multiple correspondence analysis of a crosstabulation matrix using the Kohonen algorithm, *Proc. ESANN'95*, M.Verleysen Ed., Editions D Facto, Bruxelles, 27-32, 1995.

[15] M.Cottrell, B.Girard, Y.Girard, C.Muller, P.Rousset, Daily Electrical Power Curves : Classification and Forecasting Using a Kohonen Map, *From Natural to Artificial Neural Computation, Proc. IWANN'95*, J.Mira, F.Sandoval eds., Lecture Notes in Computer Science, Vol.930, Springer, 1107-1113, 1995.

[16] M.Cottrell, E. de Bodt, E.F.Henrion, Understanding the Leasing Decision with the Help of a Kohonen Map. An Empirical Study of the Belgian Market, *Proc. ICNN'96 International Conference*, Vol.4, 2027-2032, 1996.

[17] M.Cottrell, B.Girard, P.Rousset, Forecasting of curves using a Kohonen Classification, to appear in *Journal of Forecasting*, 1998.

[18] P.Demartines, Organization measures and representations of Kohonen maps, In : J.Hérault (ed), *First IFIP Working Group 10.6 Workshop*, 1992.

[19] D.Dersch, P.Tavan, Asymptotic Level Density in Topological Feature Maps, *IEEE Tr. on Neural Networks*, Vol.6, 1, 230-236, 1995.

[20] E.Erwin, K.Obermayer and K.Shulten, Self-organizing maps : stationary states, metastability and convergence rate, *Biol. Cyb.*, 67, 35-45, 1992.

[21] E.Erwin, K.Obermayer and K.Shulten, Self-organizing maps : ordering, convergence properties and energy functions, *Biol. Cyb.*, 67, 47-55, 1992.

[22] J.A.Flanagan, Self-Organizing Neural Networks, Phd. Thesis, Ecole Polytechnique Fédérale de Lausanne, 1994.

[23] J.A.Flanagan, Self-organisation in Kohonen's SOM, *Neural Networks*, Vol. 6, No.7, 1185-1197, 1996.

[24] J.C.Fort, Solving a combinatorial problem via self-organizing process : an application of the Kohonen algorithm to the travelling salesman problem, *Biol. Cyb.*, 59, 33-40, 1988.

[25] J.C.Fort and G.Pagès, A non linear Kohonen algorithm, *Proc. ESANN'94*, M.Verleysen Ed., Editions D Facto, Bruxelles, 221-228, 1994.

[26] J.C.Fort and G.Pagès, About the convergence of the generalized Kohonen algorithm, *Proc. ICANN'94*, M.Marinero, P.G.Morasso Eds., Springer, 318-321, 1994.

[27] J.C.Fort and G.Pagès, On the a.s. convergence of the Kohonen algorithm with a general neighborhood function, *Annals of Applied Probability*, Vol.5, 4, 1177-1216, 1995.

[28] J.C.Fort and G.Pagès, About the Kohonen algorithm : strong or weak self-organisation, *Neural Networks*, Vol.9, 5, 773-785, 1995.

[29] J.C.Fort and G.Pagès, Convergence of Stochastic Algorithms : from the Kushner & Clark theorem to the Lyapunov functional, *Advances in Applied Probability*, 28, 4, 1072-1094, 1996.

[30] J.C.Fort and G.Pagès, Asymptotics of the invariant distributions of a constant step stochastic algorithm, to appear in *SIAM Journal of Control and Optimization*, 1996.

[31] J.C.Fort and G.Pagès, Quantization *vs* Organization in the Kohonen SOM, *Proc. ESANN'96*, M.Verleysen Ed., Editions D Facto, Bruges, 85-89, 1996.

[32] G.J.Goodhill, T.Sejnowski, Quantifying neighbourhood preservation in topographic mappings, *Proc. 3rd Joint Symposium on Neural Computation*, 61-82, 1996.

[33] M.Herrmann, H.-U. Bauer, T.Vilmann, Measuring Topology Preservation in Maps of Real-World Data, *Proc. ESANN'97*, M.Verleysen Ed., Editions D Facto, Bruxelles, 205-210, 1997.

[34] M.Hirsch, Systems of differential equations which are competitive or cooperative II : convergence almost everywhere, *SIAM J. Math. Anal.*, 16, 423-439, 1985.

[35] T.Kohonen, Self-organized formation of topologically correct feature maps, *Biol. Cyb.*, 43, 59-69, 1982.

[36] T.Kohonen, Analysis of a simple self-organizing process, *Biol. Cyb.*, 44, 135-140, 1982.

[37] T.Kohonen, *Self-organization and associative memory* Springer, New York Berlin Heideberg, 1984 (3rd edition 1989).

[38] T.Kohonen, Speech recognition based on topology preserving neural maps, in : I.Aleksander (ed) *Neural Computation* Kogan Page, London, 1989.

[39] T.Kohonen, Self-organizing maps : optimization approaches, in : T.Kohonen et al. (eds) *Artificial neural networks, vol. II*, North Holland, Amsterdam, 981-990, 1991 .

[40] T.Kohonen, *Self-Organizing Maps*, Vol. 30, Springer, New York Berlin Heiderberg, 1995.

[41] H.J.Kushner, D.S.Clark, *Stochastic Approximation for Constrained and Unconstrained Sysqtems*, Volume 26, in Applied Math. Science Series, Springer, 1978.

[42] S.P.LLoyd et al. *Special Issue on Quantization*, IEEE Tr. on Information Theory, Vol.IT-28, No.2, 129-137, 1982.

[43] D.Lamberton, G.Pagès, On the critical points of the 1- dimensional Competitive Learning Vector Quantization Algorithm, *Proc. ESANN'96*, M.Verleysen Ed., Editions D Facto, Bruges, 1996.

[44] Y.Linde, A.Buzo, R.Gray, *An Algorithm for Vector Quantizer Design*, IEEE Tr. on Communications, Vol. 28, No. 1, 84-95, 1980.

[45] Z.P.Lo, B.Bavarian, On the rate of convergence in topology preserving neural networks, *Biol. Cyb*, 65, 55-63, 1991.

[46] Z.P.Lo, Y.Yu and B.Bavarian, Analysis of the convergence properties of topology preserving neural networks, *IEEE trans. on Neural Networks*, 4, 2, 207-220, 1993.

[47] S.Luttrell, Derivation of a class of training algorithms, *IEEE Transactions on Neural Networks*, 1 (2), 229-232, 1990.

[48] D.J.Newman, The Hexagon Theorem, *Special Issue on Quantization, IEEE Tr. on Information Theory*, Vol.IT-28, No.2, 137-139, 1982.

[49] E.Oja, Self-organizing maps and computer vision, in : H.Wechsler (ed), *Neural networks for Perception*, vol.1, Academic Press, Boston, 1992.

[50] G.Pagès, Voronoï tesselation, space quantization algorithms and numerical integration, in *Proc. of the ESANN93 Conference*, Bruxelles, Quorum Ed., (ISBN-2-9600049-0-6), 221-228, 1993.

[51] G.Pagès, Numerical Integration by Space Quantization, Technical Report, 1996.

[52] H.Ritter and K. Schulten, On the stationary state of Kohonen's self-organizing sensory mapping, *Biol. Cybern.*, 54, 99-106, 1986.

[53] H.Ritter and K. Schulten, Convergence properties of Kohonen's topology conserving maps: fluctuations, stability and dimension selection, *Biol. Cybern.*, 60, 59-71, 1988.

[54] H.Ritter T.Martinetz and K. Schulten, Topology conserving maps for motor control, *Neural Networks, from Models to Applications*, (L.Personnaz and G.Dreyfus eds.), IDSET, Paris, 1989.

[55] H.Ritter, Asymptotic Level Density for a Class of Vector Quantization Processes, *IEEE Tr. on Neural Networks*, Vol.2, 1, 173-175, 1991.

[56] H.Ritter T.Martinetz and K. Schulten, *Neural computation and Self-Organizing Maps, an Introduction*, Addison-Wesley, Reading, 1992.

[57] H.Robbins and S. Monro, A stochastic approximation method, *Ann. Math. Stat.*, vol. 22, 400-407, 1951.

[58] P.Růžička, On convergence of learning algorithm for topological maps, *Neural Network World*, 4, 413-424, 1993.

[59] A.Sadeghi, Asymptotic Behaviour of Self-Organizing Maps with Non-Uniform Stimuli Distribution, *Annals of Applied Probability*, 8, 1, 281-289, 1997.

[60] A.Sadegui, Self-organization property of Kohonen's map with general type of stimuli distribution, submitted to *Neural Networks*, 1997.

[61] P.Thiran, M.Hasler, Self-organization of a one-dimensional Kohonen network with quantized weights and inputs, *Neural Networks*, 7(9), 1427-1439, 1994.

[62] T.Villmann, R.Der, T.Martinetz, A novel approach to measure the topology preservation of feature maps, *Proc. ICANN'94*, M.Marinero, P.G.Morasso Eds., Springer, 298-301, 1994.

[63] T.Villmann, R.Der, T.Martinetz, Topology Preservation in Self-Organizing Feature Maps: Exact Definition and Measurement, *IEEE Tr. on Neural Networks*, Vol.8, 2, 256-266, 1997.

[64] P.L.Zador, Asymptotic Quantization Error of Continuous Signals and the Quantization Dimension, *Special Issue on Quantization, IEEE Tr. on Information Theory*, Vol.IT-28, No.2, 139-149, 1982.

[65] S.Zrehen, F.Blayo, A geometric organization measure for Kohonen's map, in: *Proc. of Neuro-Nîmes*, 603-610, 1992.