Artificial Neural Networks
T. Kohonen, K. Mäkisara, O. Simula and J. Kangas (Editors)

Convergence Properties of Self-Organizing Maps

E. Erwin, K. Obermayer and K. Schulten

Beckman-Institute, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA, Email: oby@lisboa.ks.uiuc.edu

Abstract:

We investigate the convergence properties, in particular convergence time and number and characteristics of metastable states, of the self-organizing feature map algorithm for a simple, but very instructive case: the representation of the unit interval by a linear Kohonen chain. We find that convergence times are minimal for a Gaussian neighborhood function with a width of the order of the number of neurons in the chain. Metastable states, which may "trap" Kohonen maps for a long time during the ordering process, arise for concave-shaped neighborhood functions. An extension of Kohonen's proof of ordering to include all neighborhood functions which are monotonically decreasing with distance is introduced.

1. Introduction

The self-organizing feature map algorithm (SFM) [1,2] is an iterative procedure to generate a representation of an often continuous input space by a discrete set of prototypes, *weight vectors*, which are associated with points, *neurons*, in some image domain or *network*. Besides approximating the input space by these prototypes, the SFM algorithm also arranges them in such a way that the metric relationships of its elements are mirrored by the metric relationships of the points associated with the prototypes. This requires that neighboring input patterns are mapped onto neighboring neurons. The desired result is an *optimal representation, i.e.* an image (*map*) of the input space in which the most important similarity relationships among patterns are preserved and transformed into spatial neighborhood relationships on the network.

The representation of data generated by the SFM algorithm has proven useful for a variety of technical applications in the areas of pattern classification and function approximation [3–5] and has successfully been applied as a model for patterns and pattern formation in biological neural systems [6]. However, a general theory of map formation is still out of sight, and even problems of important practical interest, like the number and type of *optimal representations*, the convergence to optimal representations, convergence speed as a function of the algorithm's parameters and the avoidance of sub-optimal representations, are not solved.

The intent of this paper is to shed some light on these questions for a simple, but very instructive case: the formation of a topographic representation of the unit interval by a linear chain. We will first outline an extension of Kohonen's proof of convergence [7], and then consider the issue of the rate of convergence. It can be shown that the *rate* at which the algorithm converges depends on the shape of the so-called neighborhood function. In particular, for a fixed value of the learning step, there is an optimum value for the width of the neighborhood function, for which convergence time is the shortest. The "best" neighborhood function to use turns out to be one which is "convex" over a large range around the winner neuron, and yet which has large differences in values at neighboring neurons. For a Gaussian function, which is typically chosen, these competing interests balance when the "full width at half-height" of the Gaussian is of the order of the number of neurons in the chain. It can be proven that no *metastable* states exist for broad, convex neighborhood functions, but for Gaussian functions below a certain width, metastable states appear. These metastable states are fixed points of the mapping algorithm other than the optimal representation. The mapping algorithm may be "trapped" in these metastable states for a finite number of iterations before the optimal representation is found. For neighborhood functions which are not convex anywhere,

metastable states exist for all parameters and the ordering time is much longer than for a Gaussian function, even if both functions are similar in that their distance in function space is arbitrarily small.

2. The SFM-Algorithm

In the following we are concerned with the representation of the unit interval $[0,1]$, the *input space*, by a one-dimensional network of N neurons. We define a *state* $\vec{w}$ as a particular set of weight values w_s, and a *configuration* as the set of states which are characterized by the same order relations.

The mapping process starts by assigning random initial values to each weight vector $w_s(t)$ at $t = 0$. The generation of the map by the SFM-algorithm then follows an iterative procedure. At each iteration step a *pattern* v is chosen from the input space at random. Then the neuron r whose weight vector is closest to v is selected, and the value of each weight vector is changed according to the *update rule*

$$w_s(t+1) = w_s(t) + \varepsilon h(r,s)(v - w_s(t)) \tag{1}$$

where $h(r,s)$ is usually given by a Gaussian function of the distance between a neuron s and the *winner* neuron r:

$$h(r,s) = h(|r-s|) = \exp(-(r-s)^2/\sigma^2) \tag{2}$$

One of the aims of this paper, however, is to study the effect of using different definitions of $h(r,s)$. In particular, we will consider the property of convexity; we define the neighborhood function to be *convex* on a certain interval I, if $|s-q| > |s-r|$, $|r-q| \Longrightarrow [h(0)+h(s,q)] < [h(s,r)+h(r,q)]$ for all $|s-q|$, $|s-r|$, $|r-q|$ within interval I, and to be *concave* otherwise. (See Fig. 2.)

3. Ordered Configurations

We define an *ordered* configuration as a map of the input space $[0,1]$ which preserves the distance relations between input patterns

$$|r-s| < |r-q| \Longleftrightarrow |w_r - w_s| < |w_r - w_q| \quad \forall r,s,q \tag{3}$$

and we define an *optimal* representation as a stationary state of (1) which is ordered. For a given pattern v, equation (1) results in multiplying $(w_s - v)$ by the factor $(1-\varepsilon h(r,s))$. If (3) is fulfilled for a given map, $0 < \varepsilon$, $0 < h(r,s) < 1$, and if the neighborhood function is monotonously decreasing with $|r-s|$, this factor is positive, smaller than one, and decreases with the distance between a weight and the applied pattern. Thus ordered weights cannot change order and there is no sequence of patterns which leads to a configuration violating condition (3).

Kohonen [7], and Cottrell and Fort [8] have proven that, with a neighborhood function that is a step function, the SFM-algorithm will cause any inital set of weight values to be arranged into an ordered configuration in the limit as time goes to infinity. We have extended this proof to include the set of all monotonically decreasing, positive-valued neighborhood functions in the set of neighborhood functions that will result in an ordering of the weight values. The proof rests on the facts that for such neighborhood functions, it can be shown that the only absorbing states obey (3) and that it is always possible to find some sequence of patterns v which will cause a given mapping to develop into a map obeying (3). The proof is lengthy and details will be published elsewhere [9].

4. Ordering Time

Figure 1 shows the *ordering time*, or the average number of iterations necessary to reach an ordered configuration (3), and the corresponding standard deviations as a function of σ. Ordering time is shown for two types of neighborhood functions, a Gaussian function given by eq. (2) (Fig. 1a) and a "compressed" Gaussian function defined by $h_{comp}(r,s) = 0.9 + 0.1\, h(r,s)$, where $h(r,s)$ is given by eq. (2) (Fig. 1b). The ordering time is proportional to $1/\varepsilon$ for small ε and has a sharp minimum

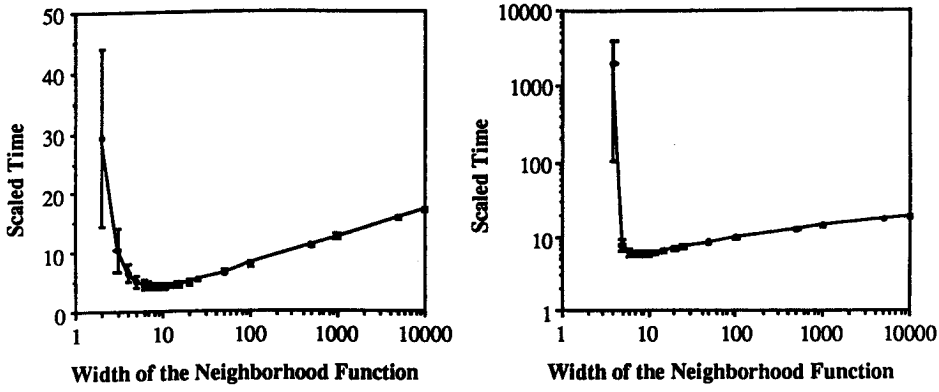

Figure 1. Average ordering time as a function of the width σ of the neighborhood function for a Kohonen chain consisting of ten neurons; a (left) Gaussian h as in (2), and b (right) "compressed" h (see text.) Time in units of $1/\epsilon$. Average is over 1000 trials. Error bars represent one standard deviation.

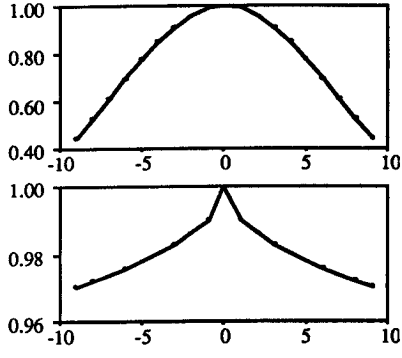

Figure 2. Examples of the neighborhood functions used: a (top) Gaussian, $\sigma = 10$; b (bottom) concave, $\sigma = 10$. (See text.)

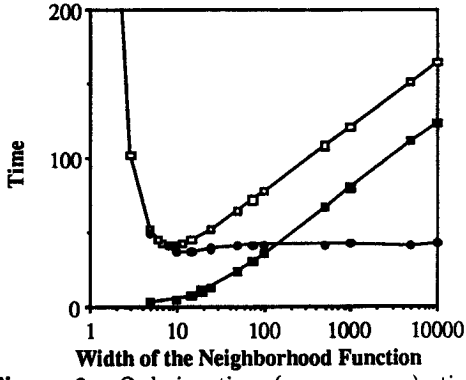

Figure 3. Ordering time (open squares), time spent in the initial configuration (closed squares), and rearrangement time (closed circles) as a function of σ for a ten-neuron Kohonen chain with a Gaussian neighborhood function ($\epsilon = 0.01$).

for $\sigma \approx 9$. The standard deviations are much larger for a small value of σ than for a large value. The parameter region around the value of $\sigma = \sigma_{min}$ which corresponds to the minimum of ordering time separates the parameter space of (1) into two regimes dominated by different phenomena which will be discussed below.

Both neighborhood functions are *convex*, at least within a certain interval of their argument, and this turns out to be important to keep ordering times short. For a "concave" neighborhood function ordering times increase dramatically. In the case of $h(r,s) = \exp(-\sqrt{|r-s|}/\sigma^2)$, and the "compressed" version $h_{comp}(r,s) = 0.9 + 0.1\ h(r,s)$, the average ordering time was even too long to be determined. This result indicates that ordering times are determined more by the "shape" of the neighborhood function than by its overall "height," and that neighborhood functions which are close (in some function space metric) can still lead to very different ordering times. Figure 2 illustrates the difference between the Gaussian and concave functions. The optimal neighborhood functions turn out to be functions which are convex for all r, s of a given Kohonen chain and for which $|h(x) - h(x+1)|$ is as large as possible.

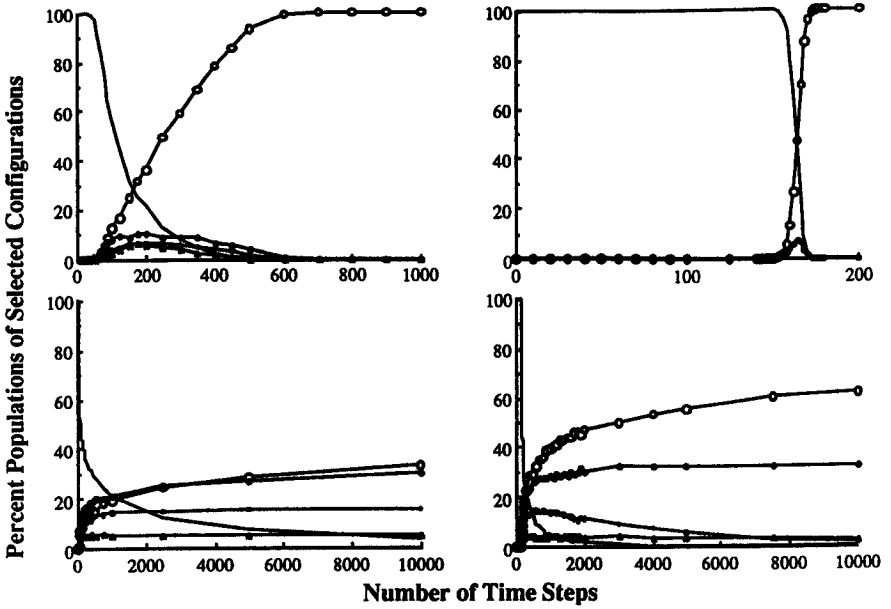

Figure 4. Percentage of ordered maps and maps in disordered configurations as a function of time for 5000 independent simulations for a Gaussian neighborhood function with $\sigma = 2$ (a, upper left), $\sigma = 10000$ (b, upper right) and a concave neighborhood function (see text) with $\sigma = 2$ (c, lower left), $\sigma = 10000$ (d, lower right). Open circles denote the percentage of maps in an ordered configuration; filled symbols, the percentage of maps in selected disordered configurations; and solid lines without plot symbol, the total percentage of maps in all other disordered configurations. The selected disordered configurations shown in a,c,d are metastable.

5. "Contraction Phase" ($\sigma >> \sigma_{min}$)

Above the optimal value σ_{min} the average ordering time increases and assymptotically approaches a logarithmic function of σ. The increase in ordering time has a simple explanation: it is completely due to the increase in the amount of time spent in the initial (random) configuration (see Fig. 3).

For large σ, the SFM-algorithm generates a map which covers only a small range in the center of the unit interval, and the average distance between weights as well as the change in their distance per iteration (1) approaches zero for $\sigma \to \infty$. Since in the initial random configuration the weights cover the unit interval completely, map formation proceeds in two steps; first the range of weight values contracts while maintaining the initial random ordering of weights, and then the weights are able to rearrange to form an ordered mapping. Figure 3 shows the average ordering time, the time spent in the initial configuration, and the rearrangement time (the difference) as a function of σ for the Gaussian neighborhood function (2). The increase in ordering time above σ_{min} is completely due to the time t_c necessary for contraction. The time t_c is given by $t_c \approx \varepsilon^{-1} \ln(l/l_c[h(r,s)])$ where l denotes the length of the interval spanned by the initial weight values and l_c denotes the length of the interval covered by the weights (*critical* length) at the time where the first rearrangements occur. The distance l_c is a functional of $h(r,s)$. Its dependence on σ determines the shape of the ordering-time curves for large σ.

Figure 4b gives another way of looking at this result. The figure shows the percentage of ordered maps and maps in disordered configurations as a function of time for 5000 independent simulations. During the contraction phase ($t = 0 - 150$) the initial configuration of the maps is preserved. When at $t \approx 150$ the length of the interval covered by the weights assumes the critical value l_c, sudden

rearrangement takes place and the maps become ordered almost at once. Note the low number of maps in intermediate configurations, nearly all of which are characterized by $w_1 < w_2 < \cdots < w_8 < w_{10} < w_9$, or one of three other symmetrically equivalent configurations. This indicates that metastable states are absent in this parameter regime of the Gaussian neighborhood function, a finding which we will discuss in more detail below.

6. Metastable States

If the probability of chosing a pattern v is $P(v)$, then the average change of the weight value w_s per iteration is given by

$$V_s = -\epsilon \int_0^1 dv\, P(v) h(r,s)(v - w_s) \tag{4}$$

where r is again the label of the winner neuron. States of particular interest are those for which $V_s = 0$ for all s, the *stationary states*, since presentation of a pattern results in no change in the weight values on the average. We will call the stationary states *stable* if they belong to ordered (i.e. absorbing) configurations, and *metastable* otherwise. For $P(v)$ =const. they are given by

$$0 = \sum_{r=1}^{N} h(r,s) \int_{\Omega_r} (v - w_s)\, dv \tag{5}$$

where N and Ω_r denote the total number of neurons and the tesselation cell of neuron r, respectively.

For $h(r,s) \equiv 1$ the only stationary state is given by $w_s = 1/2$ for all s. For more general neighborhood functions one considers the actual neighborhood function $h(r,s)$ to be the effect of a perturbation $g(r,s)$ on this trivial case, namely $h(r,s) = 1 - \epsilon g(r,s)$. It is convenient to relabel the weight values with indices that are arranged in ascending order in the input space: $x < y \rightarrow w_x < w_y$, with index values ranging from 1 to N. Let $r = \mathcal{P}(x)$ be the "old" index r which denotes the position of the corresponding neurons in the Kohonen chain. By expanding (5) in a power series around $w_x = 1/2$ up to order (ϵ^2), one can determine the stationary states. (Details will be published elsewhere.) They are given by

$$w_x = \frac{1}{2} - \frac{\epsilon}{8}\left(g_{\mathcal{P}(N)\mathcal{P}(x)} - g_{\mathcal{P}(1)\mathcal{P}(x)}\right) + \frac{\epsilon^2}{16}\left(g^2_{\mathcal{P}(N)\mathcal{P}(x)} - g^2_{\mathcal{P}(1)\mathcal{P}(x)}\right) + O(\epsilon^3) \tag{6}$$

Every permutation $\mathcal{P}(x)$ which when inserted into (6) leads to weights w_x in ascending order, describes a configuration which contains one stationary state. For a convex neighborhood function, $h(r,s) = 1 - \epsilon g(r,s)$, it can be proven that there are only two states fulfilling eq. (6) and that these states correspond to the two possible ordered configurations. Since these states are absorbing, ordering is guaranteed for convex neighborhood functions. For convex neighborhood functions there exist no metastable states; for a concave neighborhood function, however, metastable states exist for all parameter values.

Figure 5 gives an overview of the metastable states for a Kohonen chain consisting of ten neurons and a Gaussian neighborhood function. The horizontal axis represents the configuration space coordinates, the width σ is plotted on the vertical axis. Each vertical line segment denotes the range of σ over which a metastable state with a particular configuration exists. For large values of σ no metastable states are present. At $\sigma \approx 5.0245$ the first metastable states appear, characterized by the configuration $w_1 < w_2 < \cdots < w_{10} < w_9$, or one of the three configurations related by symmetry. Below $\sigma \approx 4.5289$, more metastable states come into existence. Some are only present over a short range of σ values, while others seem to persist even as σ approaches 0.

The presence of metastable states has an enormous impact on the ordering time, because these states may temporarily "trap" maps during the ordering process. It is possible to prove, however, that the maps will eventually become ordered despite the presence of metastable states [9]. Figure 4 shows the percent of maps in a particular configuration as a function of time for the Gaussian neighborhood function given by (2) (Fig. 4a,b) and the concave neighborhood function given by $h(r,s) = \exp(-\sqrt{|r-s|}/\sigma^2)$ (Fig. 4c,d). Figure 4a shows the population of maps as a function of

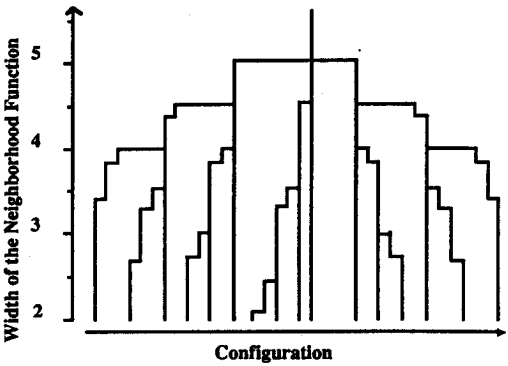

Figure 5. Bifurcation diagram showing that the number of metastable states increases as the width, σ, of the Gaussian neighborhood function is decreased. The horizontal axis is a somewhat abstract "configuration" axis. The "branch" near the center of the graph represents configurations with metastable states which are symmetric about their midpoints. The only stable configurations for $\sigma > 5.02$ are the ordered configurations. Only half of the configuration space is shown; each vertical line thus represents two states.

time for a regime where metastable states exist. Not all of the states which were predicted to be metastable are included in the figure printed here since there are too many such states. The curve that rises the highest before falling to zero represents the four symmetric configurations given by $w_1 < w_2 < \cdots < w_{10} < w_9$, and the three corresponding configurations. These states are the most "stable", i.e. the average time a map spends in one of these configurations is larger than the time spent in other metastable configurations. The curve which starts at 100% and falls to zero is the total percentage of maps in any of the configurations whose populations are not calculated separately, and may be thought of as representing the percentage of disordered maps. This behavior is very different from that in Fig. 4b which displays the result from a regime where no metastable states are present.

Figure 4c,d shows the percentage of maps in a particular configuration as a function of time for the function $h(r,s) = \exp(-\sqrt{|r-s|}/\sigma^2)$, which is concave everywhere. Metastable states exists both for large and small values of σ. Although a plot of the *concave* neighborhood function for $\sigma = 10000$ appears almost identical to a plot of a Gaussian function with $\sigma = 10000$ (particularly if one plots only the discrete points of the function,) the ordering time of the concave function is much longer than that of the Gaussian. Compare Figure 4d representing the case of a concave function to Figure 4b discussed above. Again the exact configurations whose populations are plotted are not given. The particular set of configurations which is plotted in each case is different, but in both cases the most prevalent metastable state happens to have the same configuration. However, the metastable states for small and large σ are identical for the concave neighborhood function.

Acknowledgement: The authors would like to thank H. Ritter for stimulating discussions. Financial support by the University of Illinois in a fellowship to E. E., and by the Boehringer-Ingelheim Fonds in a scholarship to K. O. is gratefully acknowledged. This research has been supported by the National Science Foundation (grant number 9017051).

References

[1] Kohonen T. (1982a), Biol. Cybern. **43**, 59-69
[2] Kohonen T. (1982b), Biol. Cybern.**44**, 135-140
[3] Kohonen T. (1989), "Speech recognition based on topology preserving neural maps", in: Aleksander I. (Ed.), Neural Computation, Kogan Page, London
[4] Nasrabadi N.M., and Feng Y. (1988), Proc. IEEE Intern. Conf. Neural Networks, Vol. I, 101
[5] Ritter H., Martinetz T., and Schulten K. (1989), Neural Networks **2**, 159-168
[6] Obermayer K., Ritter H., and Schulten K. (1990), Proc. Natl. Acad. Sci. USA **87**, 8345-8349
[7] Kohonen T. (1983), Self-Organization and Associative Memory, Springer-Verlag, New York
[8] Cottrell M., and Fort J.-C. (1987), Ann. Inst. Henri Poincare. **23**, 1-20.
[9] Erwin E., Obermayer K., and Schulten K., in preparation.