# Cellular Topographic Self-Organization under Correlational Learning

S. Sakamoto[1], S. Seki[2], and Y. Kobuchi[1]
1)Department of Electronics and Informatics, Ryukoku Univ., Seta
2)Department of Computer Science, California State Univ., Fresno

Abstract：We consider two layered binary state neural networks in which cellular topographic self-organization occurs under correlational learning. The main result is that for separable input relations, a mapping is topographic if it is stable and vice versa.

## 1. Introduction

Topographic mapping is a mapping which associates neighboring excitations at afferent cells with neighboring outputs at efferent cells. Actually, such topographic mappings as retinotopic, somatosensory, and tonotopic mappings are commonly formed in self-organized fashion at various parts of vertebrates.

We consider Willshaw-Malsburg type networks [Willshaw-Malsburg 1976] whose architecture is defined by a pair of input and output layers with connection weights. Learning scheme is based on a modified winner-take-all idea and generalized Hebb type correlational rule. In our previous works [Sakamoto and Kobuchi 2000, 2002; Sakamoto, Seki, and Kobuchi 2002] we considered two layered networks in which each input and output layer is represented by an undirected graph. A pair of cells in a layer is related when there is an edge between them in the graph representation.

Most of the previous models including ours reflect the idea of so-called local excitation inputs. We here treat a topographic mapping formation model which can treat any binary input patterns. In these frameworks, we characterize the stability of winner function under correlational learning and relate it with topographic mappings.

## 2. The Model

Let $V_I = \{1, 2, \ldots, n\}$ denote the set of input units and $V_O = \{1, 2, \ldots, m\}$, the set of output units. A synaptic weight from an input unit $j$ to an output unit $i$ is a real number between 0 and 1 given as $w_{ij}$ $[0,1]$. Then, for an output unit $i$, we have a synaptic weight vector $\mathbf{w}_i = (w_{i1}, w_{i2}, \ldots, w_{in})$. The entire synaptic weights can be represented by a weight matrix $W = [w_{ij}]$. An input pattern $X$ is a nonempty subset of $V_I$, and an input set $I$ is a non-empty set of input patterns. Each input unit $j$ $(1 \leq j \leq n)$ assumes a binary state $x_j$ $\{0,1\}$, and each input pattern $X$ determines an input vector $\mathbf{x}=(x_1, x_2, \ldots, x_n)$ by $x_k$=1 if $k$ $X$ and $x_k$=0 otherwise. We use an input pattern $X$ and the corresponding input vector $\mathbf{x}$ interchangeably. The value of an output unit $i$ $(1 \leq i \leq m)$ is a real number $y_i$ and, for an input vector $\mathbf{x}$, it is given by $y_i = \mathbf{w}_i \mathbf{x}^{\mathrm{T}}$, where $\mathbf{x}^{\mathrm{T}}$ is the transposed vector of $\mathbf{x}$.

The closeness among input (or output) units will be represented by an input (or output) neighborhood relation defined on $V_I$ (or $V_O$, respectively): $E_I \quad V_I \times V_I$ and $E_O \quad V_O \times V_O$. If $(j_1, j_2) \quad E_I$ (or $(i_1, i_2) \quad E_O$), $j_1$ and $j_2$ (or $i_1$, $i_2$) are said to be connected. The neighborhood relations $E_I$ and $E_O$ are both assumed to be reflexive and symmetric. Discarding the self loops, we can regard $(V_I, E_I)$ and $(V_O, E_O)$ as undirected graphs and call them an input graph $G_I$ and an output graph $G_O$, respectively. From these neighborhood relations, we can define an input neighborhood function $\sigma_I$ and an output neighborhood function $\sigma_O$ which, for a given unit, return its neighbors; $\sigma_I(j) = \{ k \mid (j, k) \quad E_I \}$ and $\sigma_O(i) = \{ l \mid (i, l) \quad E_O \}$. We also extend the domain of $\sigma_I$ from input units to input patterns by $\sigma_I(X) = \{ k \mid j \quad X, (j, k) \quad E_I \}$.

Now we define a network as $N = (G_I, G_O, I, \mathbf{W})$, where $I$ is an input pattern set and $\mathbf{W}$ is the set of all weight matrices. In this note, $\mathbf{W}$ is the set of all $m \times n$ matrices $[w_{ij}]$, where $w_{ij} \quad [0,1]$. With a network $N = (G_I, G_O, I, \mathbf{W})$, given a weight $W \quad \mathbf{W}$ and an input pattern $X \quad I$, we have a corresponding output vector $\mathbf{y} = (y_1, y_2, \ldots, y_m)$. Here we adopt a winner-take-all rule, that is, we consider a winner output unit from $\mathbf{y}$. For a fixed $W \quad \mathbf{W}$, this correspondence can be considered as a function $f: I \rightarrow V_O$, i.e., $f(X) = i$ where $y_i = \mathrm{Max}\{y_1, y_2, \ldots, y_m\}$. We call $f$ a winner function. In general, $f$ varies depending on $W$. Thus we have a function $F : \mathbf{W} \rightarrow V_O{}^I$. On the other hand, we can think of the set of all $W \quad \mathbf{W}$ that generate a given $f$ and will denote it as $\mathbf{W}_f$.

Let's fix $W \quad \mathbf{W}$ temporarily. When an input pattern $X \quad I$ is given, for each input unit $j$ ($1 \leq j \leq n$), we consider a binary input neighbor state $b_j \quad \{0,1\}$ which designates whether the unit is in the neighborhood of an input pattern or not: $b_j = 1$ if $j \quad \sigma_I(X)$ and $b_j = 0$ otherwise. For an output unit $i$ ($1 \leq i \leq m$), we consider similarly a binary winner neighbor state $v_i \quad \{0,1\}$ which represents whether the unit is in the neighborhood of the winner or not: $v_i = 1$ if $i \quad \sigma_O(f(X))$ and $v_i = 0$ otherwise.

Now we are ready to define the following learning scheme to change the synaptic weights in discrete time steps. If we denote relevant values at time $t$ using $t$ as a parameter, the synaptic weight at time $t+1$, $w_{ij}(t+1)$, is determined from that of time $t$, a learning rate $\alpha$ at time $t$, and a learning rule function $\delta$ by the following:

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(t)(\delta(b_j(t), v_i(t)) - w_{ij}(t))$$

where $\alpha(t)$ is a real number in (0,1) and $\delta : \{0,1\} \times \{0,1\} \rightarrow [0,1]$. The learning rule function $\delta$ represents the amount of weight changes depending on the combination of input and output state values. We mention here that the above relation can be rewritten as follows:

$$w_{ij}(t+1) = (1-\alpha(t))w_{ij}(t) + \alpha(t)\cdot\delta(b_j(t), v_i(t)).$$

Any learning rule function $\delta$ can be represented by a four-tuple of real numbers $(\delta(1,1), \delta(1,0), \delta(0,1), \delta(0,0))$. We denote the set of all learning rule functions as $\Delta$. That is, $\Delta = [0,1]^4$. We also assume that $\alpha(t)$ is a constant function, i.e., $\alpha(t)$ is fixed for any $t$ and will be written as $\alpha$. The change of the synaptic weight matrices can be considered as applying a weight matrix update function $L$ : $\mathbf{W}\times I\times\Delta\times(0,1)\rightarrow\mathbf{W}$ as follows:

$L([w_{ij}], X, \delta, \alpha) = [w'_{ij}]$, where $w'_{ij} = w_{ij} + \alpha(d_{ij} - w_{ij})$ and $d_{ij} = \delta(j \quad \sigma_I(X)$, $i \quad \sigma_O(F([w_{ij}])(X))$ where true equals 1 and false 0. That is, $L(W, X, \delta, \alpha) = W'$ means that when an input $X$ is given to the network with weight matrix $W$, it is updated to $W'$ under a learning rule $\delta$ and a learning rate $\alpha$. We call this process an $X$-learning. Geometrically speaking, an $X$-learning implies that $w_{ij}$ approaches to $\delta(b_j, v_i)$ at rate $\alpha$. When we apply a sequence of input patterns to the network, the resulting synaptic weight matrix can be computed by the following extension of $L$ : $\mathbf{W}\times I^*\times\Delta\times(0,1)\rightarrow\mathbf{W}$ defined recursively by the above together with $L(W, \lambda, \delta, \alpha) = W$ and $L(W, X_1 X_2 \ldots \quad X_r, \delta, \alpha) = L(L(W, X_1 X_2 \ldots \quad X_{r-1}, \delta, \alpha), X_r, \delta, \alpha)$.

## 3. Input Pattern Separability and Correlational Learning Rule

Let $N = (G_I, G_O, I, \mathbf{W})$ be a network. Here we introduce a reflexive relation $R_I$ over $I$. The relation is, in fact, to denote the closeness of the input patterns in $I$. That is, for $X_i \quad I$ and $X_j \quad I$, $X_i$ is considered to be close to $X_j$ if and only if $(X_i, X_j) \quad R_I$.

Definition 1. Let $N = (G_I, G_O, I, \mathbf{W})$ be a network. For any $X_i \quad I$ and $X_j \quad I$, we define $\beta_{ij}$ as follows to represent the degree of overlap between $X_i$ and $\sigma_I(X_j)$ : $\beta_{ij} = |X_i \cap \sigma_I(X_j)| / |X_i|$.

Definition 2. Let $N = (G_I, G_O, I, \mathbf{W})$ be a network. Let $\gamma \quad (0,1)$. An input pattern relation $R_I$ on $I$ is said to be $\gamma$-separable if for any $X_i, X_j \quad I$,

$(X_i, X_j) \quad R_I$ implies $\beta_{ij} > \gamma$, and $(X_i, X_j) \quad R_I$ implies $\beta_{ij} < \gamma$.

Definition 3. Let $N = (G_I, G_O, I, \mathbf{W})$ be a network. For a relation $R_I$ on $I$, let $\mu$ and $\rho$ be defined as follows.

$\mu = \mathrm{Min}\,\{\beta_{ij} \mid (X_i, X_j) \quad R_I\}$ and $\rho = \mathrm{Max}\,\{\beta_{ij} \mid (X_i, X_j) \quad R_I\}$.

These $\mu$ and $\rho$ are used to characterize $\gamma$-separability of $R_I$ as follows.

Lemma 1. Let $R_I$ be a relation over $I$ and let $\gamma \quad (0,1)$, $\mu \quad [0,1]$, and $\rho \quad [0,1]$ be real numbers as defined in Definition 3. Then we have

$R_I$ is $\gamma$-separable $\Leftrightarrow$ $\rho < \gamma < \mu$.

Now we define a class of learning rules called correlational as follows.

Definition 4. Let $N = (G_I, G_O, I, \mathbf{W})$ be a network. A learning rule $\delta : \{0,1\}^2 \rightarrow [0,1]$ is said to be correlational if $\nu_0 < 0 < \nu_1$ where $\nu_0 = \delta(0,1) - \delta(0,0)$ and $\nu_1 = \delta(1,1) - \delta(1,0)$.

## 4. X-learning and Stability of Winner Function

Let $W = [w_{lk}] \quad \mathbf{W}$ be a weight matrix where $F(W) = f$. Consider an input pattern $X_j \quad I$ and apply an $X_j$-learning to the network defined by $W$. Then, assume that we have an updated matrix $W' = L(W, X\ , \delta, \alpha)$. Each entry of $W'$ can be written as follows:

$$w'_{lk} = (1\text{-}\alpha) w_{lk} + \alpha\delta(k \quad \sigma_I(X_j),\ l \quad \sigma_O(f(X_j)))$$

Now we evaluate output value $y'_l$ at an output unit $l$ of the updated matrix $W'$ for an input pattern $X_i \quad I$.

$$\begin{aligned} y'_l &= \sum_{k \quad X_i} w'_{lk} \\ &= \sum_{k \quad X_i} \{ (1\text{-}\alpha) w_{lk} + \alpha\delta(k \quad \sigma_I(X_j),\ l \quad \sigma_O(f(X_j))) \} \end{aligned}$$

Noting that $y_l = \sum_{k \quad X_i} w_{lk}$

$$y'_l = \{ \begin{matrix} (1-\alpha) y_l + \alpha\{\delta(1,1)|X_i \cap \sigma_I(X_j)| + \delta(0,1)|X_i - \sigma_I(X_j)|\} \ if\ l \quad \sigma_O(f(X_j)) \\ (1-\alpha) y_l + \alpha\{\delta(1,0)|X_i \cap \sigma_I(X_j)| + \delta(0,0)|X_i - \sigma_I(X_j)|\} \ if\ l \quad \sigma_O(f(X_j)) \end{matrix}$$

Since $\beta_{ij} = |X_i \cap \sigma_I(X_j)| / |X_i|$, we can rewrite the above as

$$y'_l = \{ \begin{matrix} (1-\alpha) y_l + \alpha|X_i|\{\delta(1,1)\beta_{ij} + \delta(0,1)(1-\beta_{ij})\} \ if\ l \quad \sigma_O(f(X_j)) \\ (1-\alpha) y_l + \alpha|X_i|\{\delta(1,0)\beta_{ij} + \delta(0,0)(1-\beta_{ij})\} \ if\ l \quad \sigma_O(f(X_j)) \end{matrix}$$

For notational convenience, we put

$c_1 = |X_i|\{\delta(1,1)\beta_{ij} + \delta(0,1)(1-\beta_{ij})\}$ and

$c_0 = |X_i|\{\delta(1,0)\beta_{ij} + \delta(0,0)(1-\beta_{ij})\}$.

Our concern is under what condition this $X_j$-learning does not change the winner function. In other words, when $F(W') = f$ holds ? Let $W$ be any weight matrix such that $F(W) = f$. Let $W'$ be the updated matrix of $X_j$-learning in $W$. If we put $F(W') = f'$, then $f$ is $X_j$-stable when $f'(X_i) = f(X_i)$ for every $X_i \quad I$.

For an arbitrarily fixed $X_i \quad I$, we have the following cases.

If $\sigma_O(f(X_j)) = V_O$ then $y'_l = (1-\alpha) y_l + \alpha c_1$ for any $l \quad \{1, 2, \ldots, m\}$ and $f'(X_i) = $ u if $f(X_i) = $ u. That is, we have $f'(X_i) = f(X_i)$ in this case. Let $V_S = \{k \quad V_O \mid \sigma_O(k) = V_O\}$. Then if $f(X_j) \quad V_S$, $f$ is $X_j$-stable. On the other hand, if $f(X_j) \quad V_O - V_S$ then $y'_l = (1-\alpha) y_l + \alpha c_1$ when $l \quad \sigma_O(f(X_j))$ and $y'_l = (1-\alpha) y_l + \alpha c_0$ when $l \quad \sigma_O(f(X_j))$.

When $f(X_i) \quad \sigma_O(f(X_j))$, $f'(X_i) = f(X_i)$ holds if $c_1 \geq c_0$ which means

$$\delta(1,1)\beta_{ij} + \delta(0,1)(1-\beta_{ij}) \geq \delta(1,0)\beta_{ij} + \delta(0,0)(1-\beta_{ij}).$$

Similarly, when $f(X_i) \quad \sigma_O(f(X_j))$, $f'(X_i) = f(X_i)$ if $c_0 \geq c_1$ which means

$$\delta(1,0)\beta_{ij} + \delta(0,0)(1-\beta_{ij}) \geq \delta(1,1)\beta_{ij} + \delta(0,1)(1-\beta_{ij}).$$

The above inequality conditions can be rewritten as follows.

Since $c_1 - c_0 = |X_i|\{\nu_1\beta_{ij} + \nu_0(1-\beta_{ij})\}$

$$c_1 > c_0 \Leftrightarrow \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) > 0.$$

To sum up the above argument, we have the following results.

Lemma 2. After an $X_j$-learning, for any $X_i \quad I$, $f'(X_i) = f(X_i)$ holds

If $\sigma_O(f(X_j)) = V_O$ or

else if $\nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \geq 0$ when $(f(X_i), f(X_j)) \quad E_O$ or else if $\nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \le 0$ when $(f(X_i), f(X_j)) \quad E_O$.

We can show that the converse to Lemma 2 also holds true and hence we have
Theorem 3. For a network $N=(G_I, G_O, I, \mathbf{W})$, let $F(W) = f$ for $W \quad \mathbf{W}$. For $X_j \quad I$, $f$ is $X_j$–stable if and only if the followings hold: $\sigma_O(f(X_j)) = V_O$ or

For any $X_i \quad I$,

$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \ge 0$

$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \le 0$ .

Definition 5. Let $N=(G_I, G_O, I, \mathbf{W})$ be a network. A winner function $f: I \to V_O$ is said to be stable with respect to δ  Δif, for any $X_j \quad I$, $W \quad \mathbf{W}_f$, and α  (0,1), we have $F(L(W, X_j, \delta, \alpha)) = f$.

As a Corollary to Theorem 3 we have the following characterization of stable winner functions.

Corollary. $f: I \to V_O$ is stable with respect to δ  Δiff the following holds:

For $X_j \quad I$ such that $f(X_j) \quad V_O - V_S$, and for $X_i \quad I$

$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \ge 0$

$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \le 0.$

## 5.Topographic Mappings and $\gamma$ -Separable Relations

Topographic mappings are the mappings which preserve topologies of input and output spaces. In our framework, a basic definition of being topographic goes as follows.

Definition 6. $f: I \to V_O$ is said to be topographic with respect to $R_I$ and $E_O$ iff the following holds:

$X_j \quad I$ such that $f(X_j) \quad V_O - V_S$ and for $X_i \quad I$

$(X_i, X_j) \quad R_I \Leftrightarrow (f(X_i), f(X_j)) \quad E_O$

Now we are ready to prove the following main theorem of this paper.

Theorem 4. Let δ  Δbe correlational and let $R_I$ be $\nu_0/(\nu_0-\nu_1)$–separable. Then $f: I \to V_O$ is topographic iff it is stable with respect to δ  Δ

First, note the following lemma, which is a direct application of Definition 2 when $\gamma = \nu_0/(\nu_0-\nu_1)$.

Lemma 5. Let δ  Δbe correlational. Then $R_I$ is $\nu_0/(\nu_0-\nu_1)$–separable iff

$(X_i, X_j) \quad R_I \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) > 0$

$(X_i, X_j) \quad R_I \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) < 0.$

Now a proof of the main theorem is given below.

Let δ  Δbe a correlational learning rule. And let $R_I$ be $\nu_0/(\nu_0-\nu_1)$–separable.

I) Assume that $f: I \to V_O$ is topographic. Then for $X_j \quad I$ such that $f(X_j) \quad V_O - V_S$ and $X_i \quad I$

$(X_i, X_j) \quad R_I \Leftrightarrow (f(X_i), f(X_j)) \quad E_O$ by definition.

$(f(X_i), f(X_j)) \quad E_O \qquad (X_i, X_j) \quad R_I \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) > 0$

$$(f(X_i), f(X_j)) \quad E_O \qquad (X_i, X_j) \quad R_I \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) < 0$$

Then a fortiori

$$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \geq 0$$

$$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \leq 0,$$

which means $f$ is stable.

II) Assume that $f : I \to V_O$ is stable with respect to $\delta \quad \Delta$. For $X_j \quad I$ such that $f(X_j) \quad V_O - V_S$ and $X_i \quad I$

$$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \geq 0$$

$$(f(X_i), f(X_j)) \quad E_O \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \leq 0.$$

Since $\delta$ is correlational and $R_I$ is $\nu_0 /(\nu_0 - \nu_1)$ –separable, we have

$$(X_i, X_j) \quad R_I \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) > 0$$

$$(X_i, X_j) \quad R_I \qquad \nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) < 0.$$

If $\nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) > 0$ and $(f(X_i), f(X_j)) \quad E_O$ holds, then $\nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) \leq 0$ and contradiction occurs. Thus, $\nu_1\beta_{ij} + \nu_0(1-\beta_{ij}) > 0$ implies $(f(X_i), f(X_j)) \quad E_O$. That is,

$$(X_i, X_j) \quad R_I \qquad (f(X_i), f(X_j)) \quad E_O \text{ and similarly}$$

$$(X_i, X_j) \quad R_I \qquad (f(X_i), f(X_j)) \quad E_O$$

which means $f$ is topographic.

## 6. Concluding Remarks

We considered a topographic mapping formation model in Willshaw-Malsburg type networks which are less studied but seem biologically more relevant.[Van Hulle 2000] Our learning method is of generalized Hebb type with parameterized correlational scheme.

The main results are

1) If closeness relations are given, it can be used to define separability of input patterns.
2) Under correlational learning and separable input relations, a mapping is topographic if it is stable and vice versa.

Since topographic mappings can be utilized as pattern classifier, the above general results give a rigorous way to predict an asymptotic categorization of input patterns with closeness relations.

## References

Sakamoto & Kobuchi(2000) Neural Networks Vol.13, pp. 709-718

Sakamoto & Kobuchi(2002) IEICE Trans. on Information Systems Vol. E85-D No. 7 pp. 1145-1152

Sakamoto, Seki & Kobuchi(2002) Journal of Japan Society for Fuzzy Theory and Systems. Vol. 14 No. 1 pp. 43-54

Van Hulle(2000) Faithful representations and topographic maps. John Wiley & Sons

Willshaw & Malsburg (1976) Proc. Roy. Soc. Lond. B 194, pp. 431-445