1997

Objective Functions for Neural Map Formation

Laurenz Wiskott[1] and Terrence Sejnowski[123]

[1]Computational Neurobiology Laboratory
[2]Howard Hughes Medical Institute
The Salk Institute for Biological Studies, San Diego, CA 92186-5800
{wiskott,terry}@salk.edu, http://www.cnl.salk.edu/CNL

[3]Department of Biology
University of California, San Diego
La Jolla, CA 92093

Abstract

A unifying framework for analyzing models of neural map formation is presented based on growth rules derived from objective functions and normalization rules derived from constraint functions. Coordinate transformations play an important role in deriving various rules from the same function. Ten different models from the literature are classified within the objective function framework presented here. Though models may look different, they may actually be equivalent in terms of their stable solutions. The techniques used in this analysis may also be useful in investigating other types of neural dynamics.

1 Introduction

Computational models of neural map formation can be considered on at least three different levels of abstraction: detailed neural dynamics, abstract weight dynamics, and objective functions from which dynamical equations may be derived. Objective functions provide many advantages in analyzing systems analytically and in finding stable solutions by numerical simulations. The goal here is to provide a unifying objective function framework for a wide variety of models and to provide means by which analysis becomes easier. A more detailed description of this work is given in [14].

2 Correlations

The architecture considered here consists of an input layer all-to-all connected to an output layer without feed-back connections. Input neurons are indicated by ρ (retina), and output neurons by τ (tectum). The dynamics in the input layer is described by neural activities a_ρ, which yield mean activities $\langle a_\rho \rangle$ and correlations $\langle a_\rho, a_{\rho'} \rangle$. Assume these activities propagate in a linear fashion through feed-forward connections $w_{\tau\rho}$ from input to output neurons and *effective* lateral connections $D_{\tau\tau'}$ among output neurons. $D_{\tau\tau'}$ is assumed to be symmetrical and represents functional aspects of the lateral connectivity rather than the connectivity itself. We also assume a linear correlation function $\langle a_{\rho'}, a_\rho \rangle$ and $\langle a_{\rho'} \rangle$ = constant. The activity of output neurons then is $a_\tau = \sum_{\tau'\rho'} D_{\tau\tau'} w_{\tau'\rho'} a_{\rho'}$. With $i = \{\rho, \tau\}$, $j = \{\rho', \tau'\}$, $D_{ij} = D_{ji} = D_{\tau\tau'} D_{\rho'\rho} = D_{\tau\tau'} \langle a_{\rho'}, a_\rho \rangle$, and $A_{ij} = A_{ji} = D_{\tau\tau'} \langle a_{\rho'} \rangle$ we obtain mean activity and correlation

$$\langle a_\tau \rangle = \sum_{ij} A_{ij} w_j \, , \quad (1)$$

$$\langle a_\tau, a_\rho \rangle = \sum_{ij} D_{ij} w_j \, . \quad (2)$$

Since the right hand sides of Equations (1) and (2) are formally equivalent, we will discuss only the latter, which contains the former as a special case. This correlation model is accurate for linear models [e.g. 2, 5, 7, 8] and is an approximation for non-linear models [e.g. 3, 6, 10, 11, 12, 13].

3 Objective Functions

With Equation (2) a linear Hebbian growth rule can be written as $\dot{w}_i = \sum_j D_{ij} w_j$. This dynamics is curl-free, i.e. $\partial \dot{w}_i / \partial w_j = \partial \dot{w}_j / \partial w_i$, and thus can be generated as a gradient flow. A suitable objective function is $H(\mathbf{w}) = \frac{1}{2} \sum_{ij} w_i D_{ij} w_j$ since it yields $\dot{w}_i = \partial H(\mathbf{w}) / \partial w_i$.

A dynamics that cannot be generated by an objective function directly is $\dot{w}_i = w_i \sum_j D_{ij} w_j$ [e.g. 5], because it is not curl-free. However, it is sometimes possible to convert a dynamics with curl into a curl-free dynamics by a coordinate transformation. Applying the transformation $w_i = \frac{1}{4} v_i^2$ yields $\dot{v}_i = \frac{1}{2} v_i \sum_j D_{ij} \frac{1}{4} v_j^2$, which is curl free and can be generated as a gradient flow. A suitable objective function is $H(\mathbf{v}) = \frac{1}{2} \sum_{ij} \frac{1}{4} v_i^2 D_{ij} \frac{1}{4} v_j^2$. Transforming the dynamics of $\mathbf{v}$ back into the original coordinate system, of course, yields the original dynamics for $\mathbf{w}$. Coordinate transformations thus can provide objective functions for dynamics that are not curl-free. Notice that $H(\mathbf{v})$ is the same objective function as $H(\mathbf{w})$ evaluated in a different coordinate system. Thus $H(\mathbf{v}) = H(\mathbf{w}(\mathbf{v}))$ and H is a Lyapunov function for both dynamics.

More generally, for an objective function H and a coordinate transformation $w_i = w_i(v_i)$

$$\dot{w}_i = \frac{\mathrm{d}}{\mathrm{d}t} [w_i(v_i)] = \frac{\mathrm{d}w_i}{\mathrm{d}v_i} \dot{v}_i = \frac{\mathrm{d}w_i}{\mathrm{d}v_i} \frac{\partial H}{\partial v_i} = \left(\frac{\mathrm{d}w_i}{\mathrm{d}v_i} \right)^2 \frac{\partial H}{\partial w_i}, \tag{3}$$

which implies that the coordinate transformation simply adds a factor $(\mathrm{d}w_i/\mathrm{d}v_i)^2$ to the original growth term obtained in the original coordinate system. Equation (3) shows that fixed points are preserved under the coordinate transformation in the region where $\mathrm{d}w_i/\mathrm{d}v_i$ is defined and finite but that additional fixed points may be introduced if $\mathrm{d}w_i/\mathrm{d}v_i = 0$. In Figure 1, the effect of coordinate transformations is illustrated by a simple example.

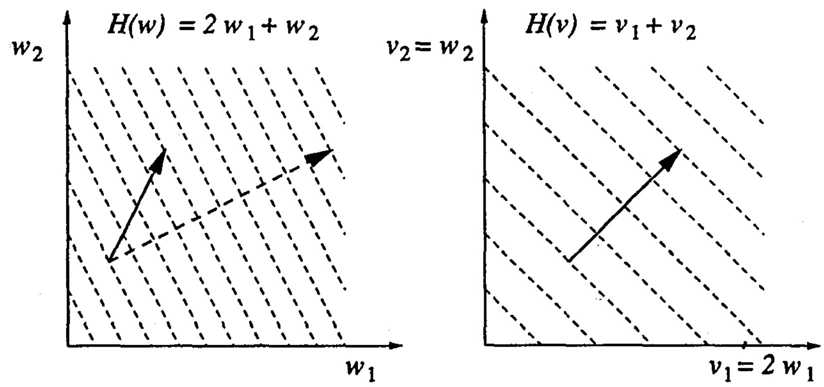

Figure 1: The effect of coordinate transformations on the induced dynamics: The figure shows a simple objective function H in the original coordinate system $\mathcal{W}$ (left) and in the transformed coordinate system $\mathcal{V}$ (right) with $w_1 = v_1/2$ and $w_2 = v_2$. The gradient induced in $\mathcal{W}$ (dashed arrow) and the gradient induced in $\mathcal{V}$ and then back-transformed into $\mathcal{W}$ (solid arrows) have the same component in the w_2-direction but differ by a factor of four in the w_1-direction (cf. Eq. 3). Notice that the two dynamics differ in amplitude and direction, but that H is a Lyapunov function for both.

Table 1 shows two objective functions and the corresponding induced dynamics terms they induce under different coordinate transformations. The first objective function, L, is linear in the weights and induces constant weight growth (or decay) under coordinate transformation $\mathcal{C}^1$. The growth of one weight does not depend on other weights. Term L can be used to differentially bias individual links, as required in dynamic link matching. The second objective function, Q, is a quadratic form. The induced growth rule for one weight includes other weights and is usually based on correlations between input and output neurons $\langle a_\tau, a_\rho \rangle = \sum_j D_{ij} w_j$, in which case it induces topography. Term Q may also be induced by the mean activities of output neurons $\langle a_\tau \rangle = \sum_j A_{ij} w_j$.

		Coordinate Transformations			
		C^1	C^α	C^w	$C^{\alpha w}$
		$w_i = v_i$ $\left(\frac{\mathrm{d}w_i}{\mathrm{d}v_i}\right)^2 = 1$	$w_i = \sqrt{\alpha_i} v_i$ $\left(\frac{\mathrm{d}w_i}{\mathrm{d}v_i}\right)^2 = \alpha_i$	$w_i = \frac{1}{4} v_i^2$ $\left(\frac{\mathrm{d}w_i}{\mathrm{d}v_i}\right)^2 = w_i$	$w_i = \frac{1}{4} \alpha_i v_i^2$ $\left(\frac{\mathrm{d}w_i}{\mathrm{d}v_i}\right)^2 = \alpha_i w_i$
Objective Functions $H(\mathbf{w})$		Growth Terms: $\dot{w}_i = ... + ...$ or $\tilde{w}_i = w_i + \Delta t(... + ...)$			
L	$\sum_i \beta_i w_i$	β_i	$\alpha_i \beta_i$	$\beta_i w_i$	$\alpha_i \beta_i w_i$
Q	$\frac{1}{2} \sum_{ij} w_i D_{ij} w_j$	$\sum_j D_{ij} w_j$	$\alpha_i \sum_j D_{ij} w_j$	$w_i \sum_j D_{ij} w_j$	$\alpha_i w_i \sum_j D_{ij} w_j$
Constraint Functions $g(\mathbf{w})$		Normalization Rules (if constraint is violated): $w_i = ...$ $\forall i \in I_n$			
$\mathrm{I}_=, \mathrm{I}_\geq$	$\theta_i - w_i$	θ_i	θ_i	θ_i	θ_i
$\mathrm{N}_=, \mathrm{N}_\geq$	$\theta_n - \sum_{j \in I_n} \beta_j w_j$	$\tilde{w}_i + \lambda_n \beta_i$	$\tilde{w}_i + \lambda_n \alpha_i \beta_i$	$\tilde{w}_i + \lambda_n \beta_i \tilde{w}_i$	$\tilde{w}_i + \lambda_n \alpha_i \beta_i \tilde{w}_i$
$\mathrm{Z}_=, \mathrm{Z}_\geq$	$\theta_n - \sum_{j \in I_n} \beta_j w_j^2$	$\tilde{w}_i + \lambda_n \beta_i \tilde{w}_i$	$\tilde{w}_i + \lambda_n \alpha_i \beta_i \tilde{w}_i$	$\tilde{w}_i + \lambda_n \beta_i \tilde{w}_i^2$	$\tilde{w}_i + \lambda_n \alpha_i \beta_i \tilde{w}_i^2$

Table 1: Objective functions, constraint functions, and the dynamics terms they induce under four different coordinate transformations. Specific terms are indicated by the symbols in the left column plus a superscript taken from the first row representing the coordinate transformation. For instance, the growth term $\beta_i w_i$ is indicated by L^w and the subtractive normalization rule $w_i = \tilde{w}_i + \lambda_n \beta_i$ is indicated by $\mathrm{N}^1_=$ (or $\mathrm{N}^1_\geq$). $\mathrm{N}^w_=$ and $\mathrm{Z}^1_=$ are multiplicative normalization rules. For the classifications in Table 2 this table has to be extended by two other methods of deriving normalization rules from constraints.

4 Constraints

A constraint is either an inequality describing a surface between valid and invalid region, e.g. $g(\mathbf{w}) = w_i \geq 0$, or an equality describing the valid region as a surface, e.g. $g(\mathbf{w}) = 1 - \sum_{j \in I} w_j = 0$. A normalization rule is a particular prescription for how the constraint has to be enforced. Thus constraints can be uniquely derived from normalization rules but not vice versa. Normalization rules can be *orthogonal* to the constraint surface or *non-orthogonal* (cf. Figure 2). Only the orthogonal normalization rules are compatible with an objective function, as illustrated in Figure 3.

The method of Lagrangian multipliers can be used to derive orthogonal normalization rules from constraints. If the constraint $g(\mathbf{w}) \geq 0$ is violated for $\tilde{\mathbf{w}}$, the weight vector has to be corrected along the gradient of the constraint function g, which is orthogonal to the constraint surface, $w_i = \tilde{w}_i + \lambda \partial g / \partial \tilde{w}_i$. The Lagrangian multiplier λ is determined such that $g(\mathbf{w}) = 0$ is obtained. If no constraint is violated, the weights are simply taken to be $w_i = \tilde{w}_i$.

Consider the effect of a coordinate transformation $w_i = w_i(v_i)$. An orthogonal normalization rule can be derived from a constraint function $g(\mathbf{v})$ in a new coordinate system $\mathcal{V}$. If transformed back into the original coordinate system $\mathcal{W}$ one obtains an in general non-orthogonal normalization rule:

$$\text{if constraint is violated:} \qquad w_i = \tilde{w}_i + \lambda \left(\frac{\mathrm{d}w_i}{\mathrm{d}\tilde{v}_i}\right)^2 \frac{\partial g}{\partial \tilde{w}_i}. \tag{4}$$

This has an effect similar to the coordinate transformation in Equation (3) (cf. Figure 4). These normalization rules are indicated by a subscript = (for an equality) or $\geq$ (for an inequality), because the constraints are enforced immediately and exactly. Orthogonal normalization rules can also be

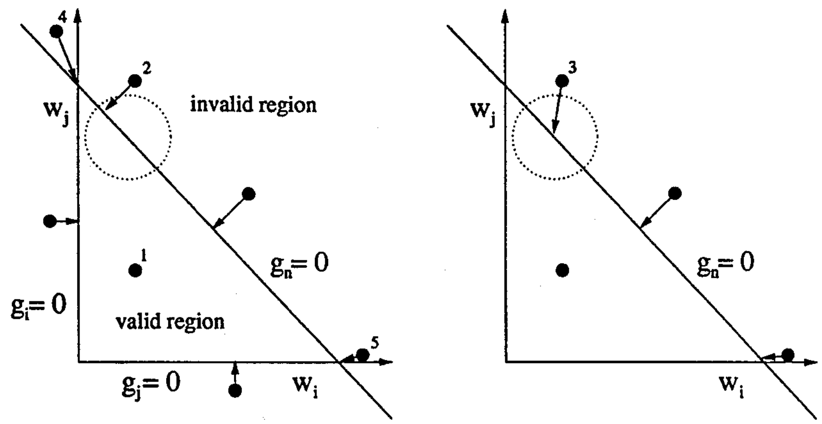

Figure 2: Different constraints and different ways in which constraints can be violated and enforced: The constraints along the axes are given by $g_i = w_i \geq 0$ and $g_j = w_j \geq 0$, which keep the weights w_i and w_j non-negative. The constraint $g_n = 1-(w_i+w_j) \geq 0$ keeps the sum of the two weights smaller or equal to 1. Black dots indicate points in state-space that may have been reached by the growth rule. Dot 1: None of the constraints is violated and no normalization rule is applied. Dot 2: $g_n \geq 0$ is violated and an orthogonal subtractive normalization rule is applied. Dot 3: $g_n \geq 0$ is violated and a non-orthogonal multiplicative normalization rule is applied. Notice that the normalization does not follow the gradient of g_n, i.e. it is not perpendicular to the line $g_n = 0$. Dot 4: Two constraints are violated and the respective normalization rules must be applied simultaneously. Dot 5: $g_n \geq 0$ is violated, but the respective normalization rule violates $g_j \geq 0$. Again both rules must be applied simultaneously. The dotted circles indicate regions considered in greater detail in Figure 3.

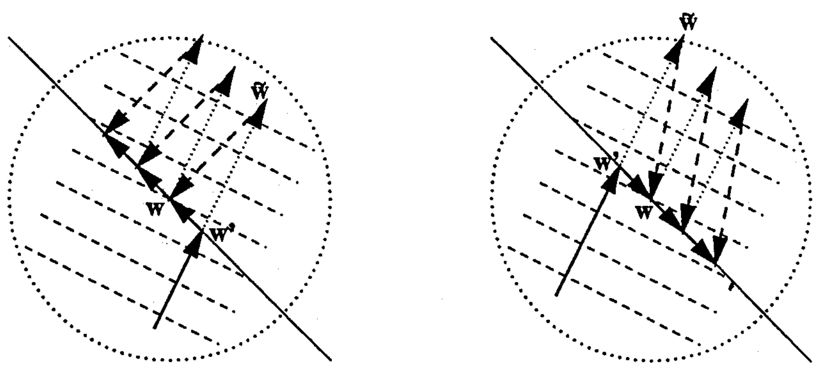

Figure 3: The effect of orthogonal versus non-orthogonal normalization rules: The two circled regions are taken from Figure 2. The effect of the orthogonal subtractive rule is shown on the left and the non-orthogonal multiplicative rule on the right. The growth dynamics is assumed to be induced by an objective function, the equipotential curves of which are shown as dashed lines. The objective function increases to the upper right. The growth rule (dotted arrows) and normalization rule (dashed arrows) are applied iteratively. The net effect is different in the two cases. For the orthogonal normalization rule the dynamics increases the value of the objective function, while for the non-orthogonal normalization the value decreases and the objective function that generates the growth rule is not even a Lyapunov function for the combined system.

Reference		Classification			
Bienenstock & von der Malsburg	[2]		Q^1	$I^1_{\geq}$	$N^1_{\approx}$
Goodhill	[3]		Q^1	$I^1_{\geq}$	$N^1_{=}$ $N^w_{=}$
Häussler & von der Malsburg	[5]	L^w	Q^w	$I^w_{>}$	$N^w_{\simeq}$
Konen & von der Malsburg	[6]		$Q^{\alpha w}$		$N^{\alpha w}_{=}$
Linsker	[7]	L^1	Q^1	$I^1_{\geq}$	
Miller et al.	[8]		Q^{α}	$I^{\alpha}_{\approx}$ $I^{\alpha}_{\geq}$	$N^{\alpha}_{=}$ ($N^w_{=}$)
Obermayer et al.	[10]		Q^1		$Z^1_{=}$
Tanaka	[11]		Q^w	$I^w_{>}$	$N^{\alpha w}_{\approx}$ ($= N^w_{\approx}$)
von der Malsburg	[12]		Q^1		$N^w_{=}$
Whitelaw & Cowan	[13]		Q^1 Q^{α}		$N^?_{=}$

Table 2: Classification of weight dynamics in previous models.

derived by other methods, e.g. penalty functions, indicated by subscripts $\approx$ and $>$, or integrated normalization, indicated by subscript $\simeq$. Table 1 shows several constraint functions and their corresponding normalization rules as derived in different coordinate systems by the method of Lagrangian multipliers. There are only two types of constraints. The first type is a *limitation constraint* I that limits the range of individual weights. The second type is a *normalization constraint* N that affects a group of weights, usually the sum, very rarely the sum of squares as indicated by Z. It is possible to substitute a constraint by a coordinate transformation. For instance, the coordinate transformation $\mathcal{C}^w$ makes negative weights unreachable and thus implements a limitation constraint $I_{>}$.

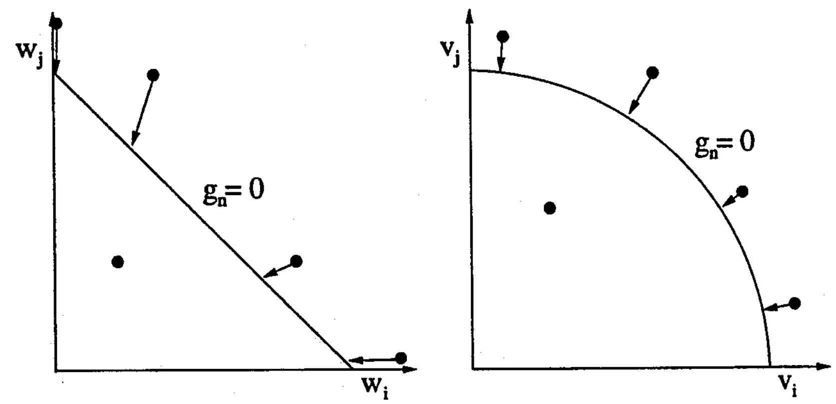

Figure 4: The effect of a coordinate transformation on a normalization rule: The constraint function is $g_n = 1 - (w_i + w_j) \geq 0$ and the coordinate transformation is $w_i = \frac{1}{4}v_i^2, w_j = \frac{1}{4}v_j^2$. In the new coordinate system $\mathcal{V}^w$ (right) the constraint becomes $g_n = 1 - \frac{1}{4}(v_i^2 + v_j^2) \geq 0$ and leads there to an orthogonal multiplicative normalization rule. Transforming back into $\mathcal{W}$ (left) then yields a non-orthogonal multiplicative normalization rule.

5 Classification of Existing Models

Table 1 summarizes the different objective functions and derived growth terms as well as the constraint functions and derived normalization rules discussed in this paper. Since the dynamics needs to be curl-free and the normalization rules orthogonal in the same coordinate system, only entries in the same column may be combined to obtain a consistent objective function framework for a system. Classifications of ten different models are shown in Table 2. The models [2, 5, 6, 7, 10] can be directly classified under one coordinate transformation. The models [3, 8, 11, 12] can probably be made consistent with minor modifications. The applicability of our objective function framework to model [13] is unclear. Another model [1] is not listed because it can clearly not be described within our objective function framework. Models typically contain three components: the quadratic term Q to induce neighborhood preserving maps, a limitation constraint I to keep synaptic weights positive, and a normalization constraint N (or Z) to induce competition between weights and to keep weights limited. The limitation constraint I can be waived for systems with positive weights and multiplicative normalization rules [6, 10, 12]. Since the model in [7] uses negative and positive weights and weights have a lower and an upper bound, no normalization rule is necessary. The weights converge to their upper or lower limit.

6 Discussion

A unifying framework for analyzing models of neural map formation has been presented. Objective functions and constraints provide a formulation of the models as constraint optimization problems. From these, weight dynamics, i.e. growth rule and normalization rules, can be derived in a systematic way. Different coordinate transformations lead to different weight dynamics, which are closely related because they usually have the same set of stable solutions. Some care has to be taken for regions where the coordinate transformation is not defined or if its derivatives become zero. We have analyzed ten different models from the literature and find that the typical system contains the quadratic term Q, a limitation constraint I, and a normalization constraint N (or Z). The linear term L has rarely been used but could play a more important role in future systems of dynamic link matching or in combination with term Q for map expansion, see below.

In addition to the unifying formalism, the objective function framework provides deeper inside into several aspects of neural map formation.

- Functional aspects of the quadratic term Q can be easily analyzed. For instance, if $D_{\rho\rho'}$ and $D_{\tau\tau'}$ are positive Gaussians, Q leads to topography preserving maps and has the tendency to *collapse*, i.e. if not prevented by individual normalization rules for each output neuron, all links coming from the input layer eventually converge on one single output neuron. The same holds for the input layer. If $D_{\rho\rho'}$ is a negative constant and $D_{\tau\tau'}$ is a positive Gaussian and in combination with a positive linear term L, topography is ignored and the map is *expanding*, i.e. even without normalization rules, each output neuron eventually receives the same total sum of weights. More complicated effective lateral connectivities can be superimposed from simpler ones.
- Because of the possible expansion effect of L + Q it should be possible to define a model without any constraints.
- The same objective functions and constraints evaluated under different coordinate transformations provide different weight dynamics that may be equivalent with respect to the stable solutions they can converge to. This is because stable fixed points are preserved under coordinate transformations with finite derivatives.
- In [9] a clear distinction between multiplicative and subtractive normalization was made. However, the concept of equivalent models shows that normalization rules have to be judged in combination with growth rules, e.g. $N^w + I^w + Q^w$ (multiplicative normalization) is equivalent to $N^1 + I^1 + Q^1$ (subtractive normalization).
- Models of dynamic link matching [2, 6] introduced similarity values rather implicitly. A more direct formulation of dynamic link matching can be derived from the objective function L + Q.
- Objective functions provide a link between neural dynamics and algorithmic systems. For instance, the C-measure proposed in [4] as a unifying objective function for many different map formation algorithms is a one-to-one mapping version of the quadratic term Q.

The objective function framework provides a basis on which many models of neural map formation can be analyzed and understood in a unified fashion. Furthermore, coordinate transformations as a tool to derive objective functions for dynamics with curl, to derive non-orthogonal normalization rules, and to unify a wide range of models might also be applicable to other types of models, such as unsupervised learning rules, and provide deeper insight there as well.

Acknowledgments

We are grateful to Geoffrey J. Goodhill, Thomas Maurer, and Jozsef Fiser for carefully reading the manuscript and useful comments. Laurenz Wiskott has been supported by a Feodor-Lynen fellowship by the Alexander von Humboldt-Foundation, Bonn, Germany.

References

[1] AMARI, S. Topographic organization of nerve fields. *Bulletin of Mathematical Biology 42* (1980), 339–364.

[2] BIENENSTOCK, E., AND VON DER MALSBURG, C. A neural network for invariant pattern recognition. *Europhysics Letters 4*, 1 (1987), 121–126.

[3] GOODHILL, G. J. Topography and ocular dominance: A model exploring positive correlations. *Biol. Cybern. 69* (1993), 109–118.

[4] GOODHILL, G. J., FINCH, S., AND SEJNOWSKI, T. J. Optimizing cortical mappings. In *Advances in Neural Information Processing Systems* (Cambridge, MA, 1996), D. Touretzky, M. Mozer, and M. Hasselmo, Eds., vol. 8, MIT Press, pp. 330–336.

[5] HÄUSSLER, A. F., AND VON DER MALSBURG, C. Development of retinotopic projections — An analytical treatment. *J. Theor. Neurobiol. 2* (1983), 47–73.

[6] KONEN, W., AND VON DER MALSBURG, C. Learning to generalize from single examples in the dynamic link architecture. *Neural Computation 5*, 5 (1993), 719–735.

[7] LINSKER, R. From basic network principles to neural architecture: Emergence of orientation columns. *Ntl. Acad. Sci. USA 83* (1986), 8779–8783.

[8] MILLER, K. D., KELLER, J. B., AND STRYKER, M. P. Ocular dominance column development: Analysis and simulation. *Science 245* (1989), 605–245.

[9] MILLER, K. D., AND MACKAY, D. J. C. The role of constraints in Hebbian learning. *Neural Computation 6* (1994), 100–126.

[10] OBERMAYER, K., RITTER, H., AND SCHULTEN, K. Large-scale simulations of self-organizing neural networks on parallel computers: Application to biological modelling. *Parallel Computing 14* (1990), 381–404.

[11] TANAKA, S. Theory of self-organization of cortical maps: Mathematical framework. *Neural Networks 3* (1990), 625–640.

[12] VON DER MALSBURG, C. Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik 14* (1973), 85–100.

[13] WHITELAW, D. J., AND COWAN, J. D. Specificity and plasticity of retinotectal connections: A computational model. *J. Neuroscience 1*, 12 (1981), 1369–1387.

[14] WISKOTT, L., AND SEJNOWSKI, T. Objective functions for neural map formation. Tech. Rep. INC-9701, Institute for Neural Computation, University of California, San Diego, La Jolla, CA 92093, Jan. 1997. 27 pages.