Field Computation in Motor Control

Bruce MacLennan*

Computer Science Department
University of Tennessee, Knoxville
maclennan@cs.utk.edu

Revised

October 30, 1996

Abstract

Field computation deals with information processing in terms of *fields*, continuous distributions of data. Many neural phenomena are conveniently described as fields, including neuron activity from large (brain area) to small (dendritic) scales. Further, it is often useful to describe motor control and sensorimotor coordination in terms of external fields such as force fields and sensory images. We survey the basic concepts of field computation, including both feed-forward field operations and field dynamics resulting from recurrent connections. Adaptive and learning mechanisms are discussed briefly. The application of field computation to motor control is illustrated by several examples: external force fields associated with spinal neurons (Bizzi & Mussa-Ivaldi 1995), population coding of direction in motor cortex (Georgopoulos 1995), continuous transformation of direction fields (Droulez & Berthoz 1991a), and linear gain fields and coordinate transformations in posterior parietal cortex (Andersen 1995). Next we survey some field-based representations of motion, including direct, Fourier, Gabor and wavelet or multiresolution representations. Finally we consider briefly the application of these representations to constraint satisfaction, which has many applications in motor control.

*This paper will appear in *Self-Organization, Computational Maps and Motor Control*, ed. by Pietro G. Morasso and Vittorio Sanguineti, Elsevier-North Holland, in press.

1 Motivation

My purpose in this chapter is to introduce the general concepts of field computation and to describe some possible applications of it to motor control. Field computation deals with continuous distributions of activity such as are found in the topographic maps and other functional areas of the brain (Knudsen et al. 1987), but also with external distributions of quantity, such as force fields. In field computation we are generally concerned with the topology of the space over which a quantity is distributed; this contrasts with the common approach in neural network modeling, which treats neural activity as a vector, that is, as quantity distributed over a space with no significant topology (since the axes are independent and, in effect, all equally distant from each other).

After defining fields and surveying their occurrence in the brain, I will give a brief introduction to the mathematics of field computation and then consider several problems in motor control from the perspective of field computation.

2 Fields

2.1 Definition

For the purposes of field computation, a *field* is defined to be *a spatially continuous distribution of quantity. Field computation* is then a computational process that operates on an entire field in parallel. Often we treat the field as varying continuously in time, although this is not necessary.

It is sometimes objected that distributions of quantity in the brain are not in fact continuous, since neurons and even synapses are discrete. However, this objection is irrelevant. For the purposes of field computation, it is necessary only that the number of units be sufficiently large that it may be treated as a continuum, specifically, that continuous mathematics can be applied. There is, of course, no specific number at which the ensemble becomes "big enough" to be treated as a continuum; this is an issue that must be resolved by the modeler in the context of the use to which the model will be put. However, since there are 146 000 neurons per mm^2 throughout most of the cortex (Changeux 1985, p. 51), it is reasonable to say that activity in a region of cortex more than a square millimeter in size can be safely treated as a field.

Mathematically, a field is treated as a continuous, usually real-valued, function ϕ over some continuum Ω, its *domain* or *extent*. For example, if Ω is a circular disk representing the retina, then for any point $p \in \Omega$, $\phi(p)$ might be the light intensity at p. The field's domain has some *topology* (relations of connectivity and nearness); for example, the topology of the retina is a two-dimensional continuum.

2.2 Realization in the Brain

There are several levels of neural activity that can be viewed as field computation.

The most obvious fields, which are measured by multiple electrode recording or by noninvasive imaging, such as NMR, are those comprising the spiking activity of neurons. Since, as we have seen, there are 146 thousand neurons per square millimeter of cortex, regions of cortex of this size are more than big enough to be treated as continua (reasonably, a tenth of a square millimeter is more than large enough). Indeed, Knudsen et al. (1987) observe that computational maps in the brain may be as small as a square millimeter, and perhaps smaller.

In cortical regions where the information is represented by impulse rate, the field is real-valued; thus $\phi(p,t)$ or $\phi_p(t)$ represents the instantaneous impulse rate at location p and time t. Recently Hopfield (1995) has argued that information may be represented by a combination of impulse frequency and phase (relative to a global "clock" field or to other neurons); in some cases at least, the phase represents an analog value and the amplitude represents its importance. In such cases it's natural to treat the field as complex-valued, with the complex number's phase angle representing the impulse phase and its magnitude representing the impulse amplitude. Thus we write $\phi_p(t) = a_p(t)e^{i\theta_p(t)}$, where $a_p(t)$ is the time-varying amplitude and $\theta_p(t)$ the time-varying phase. Synapto-dendritic transmission of such a field, which affects both its amplitude and phase, can be represented as multiplication by a constant complex number. For example, suppose a field $\psi = z\phi$ results from transmitting field ϕ through synapses $z_p = w_p e^{i\zeta_p}$ that introduce amplitude change w_p and phase shift ζ_p. Then,

$$\psi_p(t) = [w_p e^{i\zeta_p}]\ a_p(t)e^{i\theta_p(t)} = [w_p a_p(t)]e^{i[\theta_p(t)+\zeta_p]}.$$

More compactly, $\psi = (we^{i\zeta})(ae^{i\theta}) = (wa)e^{i(\zeta+\theta)}$. This encoding allows the soma potential to combine both the analog values and the importance of signals arriving at the synapses.

At the next level down we can consider the synaptic fields associated with one neuron or a group of neurons. For example, $\phi_p(t)$ represents the time-varying activity (measured, for example, by presynaptic potential or by neurotransmitter flux across the synapse) of synapse p. Certainly a pyramidal cell with 200 thousand synapses on its dendritic tree can be said to have a synaptic field, and even neurons with smaller numbers of inputs can treated as processing fields. The topology underlying the field is determined by the dendritic tree, so in many cases the synaptic field cannot be treated separately from the dendritic field (discussed next).

When we view the neuron at the level of the dendritic fields, we are concerned with the time-varying electrical potential field over the dendritic membrane. This varies continuously from point to point on the membrane and is determined by the detailed morphology of the dendritic tree. To a

first approximation, field computation in the dendritic tree can be treated as a linear system (MacLennan 1993).

Finally, there are fields at larger scales. For example, the phase delays discussed by Hopfield (1995) may be relative to "the phase of an oscillating field potential" in an area (Ferster & Spruston 1995). Further, there are global brain rhythms (α, θ etc.).

All the preceding fields are dynamic, changing on times scales of milliseconds or faster. It is often worthwhile to consider fields that are static or that change on slower time scales (for example, through learning or adaptation). Such fields are represented in the connectivity patterns between neurons and in patterns of synaptic efficacy. For example, suppose that a topographic map A projects to a topographic map B in such a way that the activity $\psi_u(t)$ of a neuron at location u in B depend on the activities $\phi_v(t)$ of neurons at locations v in A, and that the strength of the dependence is given by K_{uv}. In the simplest case we have a linear dependence,

$$\psi_u(t) = \int_\Omega K_{uv}\phi_v(t)\mathrm{d}v,$$

which we may write as a field equation, $\psi(t) = K\phi(t)$. The "kernel" K of this operator defines a connectivity field between A and B.

2.3 Reduction of Dimension

The cortex can directly represent "two-and-one-half dimensional" axonal fields. By "two-and-one-half dimensional" we mean a discrete stack of two-dimensional continua; for example, we might have six continua corresponding to six layers in the cortex. (Although synaptic and dendritic fields are embedded in three-dimensional space, the complex structure of the dendritic tree gives them a more complex non-Euclidean topology, therefore the notion of dimension is not directly applicable to them.) Some fields are naturally two dimensional, for example, a light intensity field over the retina or a pressure field over the skin.

There are many cases where the cortex must represent fields defined over more than two dimensions. For example, since cells in VI are selective for orientation ζ as well as retinal position (r, θ), the activity fields are naturally three-dimensional, $\phi(r, \theta, \zeta)$. Furthermore, there is substantial evidence (surveyed, for example, in MacLennan 1991) that they are sensitive to spatial frequency f as well, so we actually have four-dimensional fields $\phi(r, \theta, \zeta, f)$.

In these cases, representation in the cortex requires that the field be reduced to two dimensions in a way that does as little violence to the proximity relations as possible. The simplest way to do this is to "slice" the field, as we might slice a pepperoni, and arrange the pieces in a plane. More generally, the field must be cut into "nearly two-dimensional" parts that can then be arranged systematically in a plane. This is one reason for the striate and columnar structure found in many brain areas.

Non-Euclidean fields are found in neuropil (the dense nets comprising the tangled dendritic trees of many neurons) and other places where the pattern of connections alters the effective distance between points of activity. Such fields may be defined over spaces with unusual (e.g. nonmetric) topologies since, for example, the distance a signal must travel in going from A to B may be different from the distance from B to A.

2.4 External Fields Relevant to Motor Activity

Not all the fields of interest are in the brain. When investigating motor activity we also have to consider the musculo-skeletal system as well as fields external to the animal. Further, for sensory-motor coordination we have to include various sensory fields (e.g., visual, proprioceptive, auditory, vestibular). Here I'll look briefly at three examples (discussed in more detail in section 6).

First, premotor circuits in the frog spinal column have associated convergent force fields in the vicinity of the frog's leg; the activation of multiple circuits creates a linear superposition (sum) of these fields, and the resulting convergent force field guides the leg to a fixed destination independently of its current position (Bizzi & Mussa-Ivaldi 1995). This is a kind of field computation, except that the purpose is not the computation of abstract quantities, but the generation of concrete physical forces. Nevertheless, the mathematics of field computation can be used to describe and analyze the motor system.

One way to understand (nondiscursive) action planning is in terms of *environmental potential fields*, an approach which has been useful in both robotics (e.g., Khatib 1986, Rimon & Koditschek 1989) and neuroscience (e.g., Hogan 1984). In moving from one place to another we naturally select a path that minimizes some notion of work. We avoid obstacles, of course, and generally try to have a minimum path length, but this strategy may be modified by judgments of the ease of passage, etc. For example, we may go around a hedge even though the shortest path is through it; the path around minimizes work (broadly defined). Our knowledge of a region of space can be represented by a potential field in which the height of the potential at a location reflects the difficulty in going through that location. As will be described later, field operations can be used to find (in parallel) an inexpensive path through the potential field, and to revise the path dynamically if the potential field is discovered to be inaccurate (e.g. we find a large mud puddle in our path).

The potential field is not limited to encoding environmental difficulty; it can also represent internal constraints, such as the range or facility of motion of joints and limbs. Further, the potential field can be defined over nonspatial continua, to allow planning paths through more abstract "spaces."

Finally, Sanger (submitted) has explained how neural population codes can be interpreted in terms of *conditional probability density fields* (CPDFs)

defined over possible stimuli. Each neuron has a CPDF that corresponds to its receptive field; the CPDF of a population over s short time interval is given by the product of the CPDFs of the neurons firing in that interval.

3 Field Operations

3.1 Definition

The primary defining feature of field computation is that it operates on an entire field in parallel. For example, operations that process a retinal image in parallel, or which generate a spatial or motor map in parallel, are clear examples of field computation. On the other hand, a process that generates one or a few scalar signals sequentially in time is not considered field computation (except in a degenerate or trivial sense). The point is not to have a clear and absolutely precise demarcation between field computation and non-field computation — it is fundamentally a matter of degree — but to distinguish field computation *as a style of computation* from computation that is scalar or low-dimensional. The operational criterion is the ability to apply continuous mathematics to the spatial distribution of quantity.

In this section we consider *field operations*, which are commonly implemented by nonrecurrent or feed-forward connections between brain areas. That is, a pattern of activity $\phi(t)$ over an area A at time t causes a pattern of activity $\psi(t') = F[\phi(t)]$ over an area B at a slightly later time t'. More generally, activity pattern $\psi(t)$ over region B depends on earlier activity patterns $\phi_1, \ldots, \phi_n$ over regions $A_1, \ldots, A_n$:

$$\psi(t) = F[\phi_1(t - \delta_1), \ldots, \phi_n(t - \delta_n)],$$

where $\delta_1, \ldots, \delta_n$ are fixed delays. Field operations may be classified as *linear* (including *multilinear*) or *nonlinear*.

3.2 Linear Operations

A process is linear when its response to a superposition of inputs is the superposition of its response to the inputs taken separately, $L(\phi_1+\phi_2+\cdots+\phi_n) = L\phi_1 + L\phi_2 + \cdots + L\phi_n$. We must remark that there can be no purely linear processes in the nervous system, for if there were, it would mean that a response to twice the input is always twice the response to the single input, and likewise for any amplification of the input, $L(c\phi) = cL\phi$. This cannot happen, for neurotransmitters become depleted, the firing rates of neurons are limited by their refractory period, etc. Therefore, processes in the nervous system are at best *saturating linear*, that is, approximately linear until nonlinear saturation effects begin to dominate. In neuroscience, linearity is always an approximation, adopted for its mathematical convenience.

3.2.1 Domain Coordinate Transformation One of the simplest linear transformations is a *domain coordinate transformation*, which are usually implemented by the anatomical pattern of projections from one area to another. These operations transform the coordinates of the field's domain, thus distorting the shape of the field, perhaps for some information processing end or for a more efficient allocation of "neural real estate." (An example, the "logmap transformation" in the primary visual cortex, is discussed below.)

In general, if $h : A \longrightarrow B$ is a mapping from coordinates in region A to coordinates in region B, then the activity field ψ defined over B, which is induced by activity field ϕ over A, is given by $\psi \circ h = \phi$, that is, for any coordinates $p \in A$, $\psi[h(p)] = \phi(p)$. Thus, if we ignore scaling of amplitudes, the activity induced by the projection at $h(p)$ in B is equal to the source activity at p in A. Most such coordinate transformations are "one-to-one and onto," in which cases we can define the induced activity field directly: $\psi = \phi \circ h^{-1}$, or

$$\psi(q) = \phi[h^{-1}(q)]$$

for all $q \in B$. That is, the activity at q in B is given by the activity at $h^{-1}(q)$ in A. (Note that the field transformation from ϕ to ψ is linear even if the coordinate transformation h is not.)

For example, a coordinate transformation, the *logmap transformation* (Baron 1987, pp. 181–186), takes place between the retina and its first projection in the primary visual cortex (VI). If retinal coordinates are represented by a complex number z in polar coordinates (giving an angle and distance from the center of the retina), then the field ψ in VI is related to the retinal field ϕ by

$$\psi(z) = \phi(e^z),$$

where e^z is the complex exponential function. The effect of this is $\psi(\log r, \theta) = \phi(r, \theta)$, that is, radial distance is transformed logarithmically.

In addition to devoting more "neural real estate" to the center of the retina, this transformation has the effect of converting rotations and scale changes of centered images into simple translations (Schwartz 1977, Baron 1987, ch. 8). To see this, note that if $\phi'(z) = \phi(sz)$ is a scaled version of ϕ, then the corresponding VI field is

$$\psi'(\log z) = \phi'(z) = \phi(sz) = \psi(\log sz) = \psi[(\log s) + (\log z)],$$

which is $\psi(\log z)$, the image of ϕ, translated by $\log s$. Similarly, if $\phi'(z) = \phi(e^{i\theta} z)$ is a rotation of ϕ through angle θ, then the corresponding field is

$$\psi'(\log z) = \phi'(z) = \phi(e^{i\theta} z) = \psi[\log(e^{i\theta} z)] = \psi[i\theta + \log z],$$

which is $\psi(\log z)$, the image of ϕ, translated by θ (in a perpendicular direction to the other translation).

3.2.2 Representation in an Orthogonal Basis Most of the linear operators of interest to neuroscience can be computed efficiently by neural networks.[1] This is because such operators have an *orthonormal set of eigenfunctions* $\varepsilon_1, \varepsilon_2, \ldots$ with associated *eigenvalues* $\lambda_1, \lambda_2, \ldots$. Therefore the operator can be written as a summation:

$$L\phi = \sum_k \lambda_k (\varepsilon_k \cdot \phi)\varepsilon_k,$$

a procedure we call *factoring a linear operator through a discrete space.* This is an infinite sum, but there are only a finite number of eigenvalues greater than any fixed bound, so that the operator can be approximated by finite sums. The computation $\psi = L\phi$ is accomplished in two steps. In the first, inner products are formed between the input field and each of the eigenfunctions ε_k yielding a finite-dimensional vector $\mathbf{c}$, given by $c_k = \varepsilon_k \cdot \phi$. Each of these inner products could, in principal, be computed by a single neuron. This step effectively represents the input in a finite-dimensional vector space, that is, in a space with no significant topology (i.e., the axes are independent, none are nearer to each other than to the others). In the second step, the computed coefficients are used to amplitude-modulate the generation of fixed fields (specifically, the eigenfunctions), which are superposed to yield the output field: $\psi = \sum_k c_k \lambda_k \varepsilon_k$. This computation, likewise, can be computed by a single layer of neurons.

Even if the eigenfunctions of the operator are not known, in practical cases the operator can still be factored through a discrete space, since it can be computed via a finite-dimensional representation in terms of any orthonormal basis for the input space. First compute the coefficients by inner products with the basis functions, $c_k = \beta_k \cdot \phi$ (accomplished by neurons with receptive fields β_k). A finite-dimensional matrix product, $\mathbf{d} = M\mathbf{c}$ is computed by a single-layer neural network with fixed interconnection weights:

$$M_{jk} = \beta_j \cdot L\beta_k.$$

Again, topological relations between the vector and matrix elements are not significant, so there are few constraints on their neural arrangement. The output is a superposition of basis functions weighted by the computed d_j, $\psi = \sum_j d_j \beta_j$ (accomplished by neurons with output weight patterns β_j).

Computing the linear operator by means of the low-dimensional space spanned by the basis functions avoids the biologically unrealistic dense (all-to-all) connections implicit in the direct computation of the operator: $\psi_x = \int_\Omega L_{xy}\phi_y \mathrm{d}y$. (The preceding results are easily extended to the case where the input and output spaces have different basis fields.)

3.3 Multilinear Operations

Multilinear operations are functions of two or more arguments that are linear in each of their arguments separately. The most common multilinear

[1] Specifically, they are *Hilbert-Schmidt operators*, to which the following remarks apply.

operations are *bilinear*, that is, linear in each of two arguments. Again, no biological process can be purely multilinear, since its linearity must be limited by saturation and other consequences of the biology.

3.3.1 Convolution and correlation Two closely-related bilinear operations that are especially important for information processing are *convolution* and *correlation*. In the simplest case, correlation can be described as a comparison of two fields at all possible relative positions. More specifically, if χ is the correlation of two one-dimensional fields ϕ and ψ, $\chi = \phi \star \psi$, then $\chi(r)$ reflects how well ϕ and ψ match (in an inner-product sense) when relatively displaced by r.[2] Mathematically,

$$\chi(r) = \int_\Omega \phi(s-r)\psi(s)\mathrm{d}s. \tag{1}$$

Higher dimensional correlations are the same, except that r is a relative displacement *vector* rather than a scalar.

Convolution, $\chi = \phi \otimes \psi$, is essentially the same as correlation, except that the field ϕ is reflected before the comparison takes place:

$$\chi(r) = \int_\Omega \phi(r-s)\psi(s)\mathrm{d}s. \tag{2}$$

Convolution is useful because: (1) its algebraic properties are more like multiplication, and thus more familiar, than correlation; and (2) many physical processes (e.g. linear systems, such as dendritic nets) perform convolutions.

3.3.2 Pattern Manipulation One reason correlation and convolution are of interest is that they can be used for pattern recognition and generation. For example, the correlation $\phi\star\psi$ will have peaks wherever the pattern ϕ occurs in field ψ (or vice versa); occurrences of patterns less similar to ϕ (in an inner-product sense) will cause lesser peaks. Thus correlation $\phi \star \psi$ returns an activity pattern representing the spatial distribution in ψ of fields resembling ϕ.

This operation is approximately reversible. Suppose that γ is a radial field, such as a Gaussian, with a single narrow, sharp maximum. Convolving γ with a pattern ψ has the effect of blurring ψ by γ (i.e. smoothing ψ by a window of shape γ):

$$(\gamma \otimes \psi)(s) = \int_\Omega \gamma(s-u)\psi(u)\mathrm{d}u.$$

Further, if γ is first displaced by r, then the effect of the convolution is to blur ψ and displace it by r:

$$(T_r\gamma) \otimes \psi = T_r(\gamma \otimes \psi).$$

[2]Correlation can be defined relative to other kinds of transformation besides displacement, and to other measures of similarity besides the inner product; see MacLennan (1994) for details.

[The T_r operation *translates* (displaces) a field by r: $T_r\phi(s) = \phi(s-r)$.] Finally, since convolution is bilinear, if ϕ is a field containing a number of sharp peaks at various displacements r_k, then $\phi \otimes \psi$ will produce a field containing blurred copies of ψ at corresponding displacements:

$$\phi \otimes \psi = \left(\sum_k T_{r_k}\gamma\right) \otimes \psi = \sum_k (T_{r_k}\gamma) \otimes \psi = \sum_k T_{r_k}(\gamma \otimes \psi).$$

(The convolution of a superposition is a superposition of the convolutions.) Such an operation could be used for constructing a representation of the environment for motion planning. For example, if ψ is the shape of an obstacle retrieved from memory, and ψ is a map of the location of obstacles of this kind in the environment, then $\phi \otimes \psi$ represents the approximate boundaries of such obstacles in the environment.

3.3.3 Convolution Connections Since convolution and correlation are *bilinear* operators, that is, linear in each of their arguments, if one of the arguments is relatively fixed (as it would be, for example, when a sensory signal is correlated with a learned pattern), the operator is linear in its other argument: $\phi \star \psi = L\psi$ for fixed ϕ. Patterns of neural connectivity are often equivalent to a convolution or correlation with a fixed field. For example, the dependence of the activity at B_u on the activity at A_v might fall off as some simple function (e.g. Gaussian) of the distance between u and v, or as some more complex (e.g. nonsymmetrical) function of the relation between u and v. In the former case we have a *radial* connectivity field $\kappa(\|v-u\|)$, in the latter a *connectivity kernel* $\kappa(v-u)$. In either case, the contribution of region A to the activity at B_u can be written $\int_\Omega \kappa_{v-u}\phi_v(t)\mathrm{d}v$. Therefore, the field $\psi(t)$ contributed to B by A is defined

$$\psi_u(t) = \int_\Omega \kappa_{v-u}\phi_v(t)\mathrm{d}v,$$

which is $\kappa \otimes \phi(t)$, the convolution of the (unvarying) connectivity kernel κ with the activity field $\phi(t)$.

Viewing such connectivity patterns as convolutions may illuminate their function. For example, by the "convolution theorem" of Fourier analysis, the convolution $\psi = \kappa \otimes \phi(t)$ is equivalent to the multiplication $\Psi(t) = K\Phi(t)$, where $\Psi(t)$ and $\Phi(t)$ are the Fourier transforms (over the space domain) of the activity fields and K is the Fourier transform of the connectivity kernel. Thus $\Phi(t)$ represents the spatial frequency spectrum, at time t, of activity in region A, and K represents a (comparatively unvarying) spatial frequency "window" applied to this activity by its connectivity to B. For example, if κ is a Gaussian, then K is also Gaussian, and the effect of the connections is *spatial* low-pass filtering of the activity in A.

Many linear operators on fields can be approximated by convolutions implemented by neural connectivity. We will illustrate this with one useful

operator, the derivative. Suppose we have a one dimensional field ϕ and we want to compute its derivative $\psi = \phi'$. It happens that the derivative can be written as a convolution with the derivative of the Dirac delta function[3] (MacLennan 1990): $\phi' = \delta' \otimes \phi$. Like the Dirac delta, its derivative is not physically realizable, but we can compute an approximation that is adequate for neural computation. To see this, suppose that we low-pass filter ϕ before computing its derivative; this is reasonable, since the frequency content of ϕ is limited by neural resolution. In particular, suppose we filter ϕ by convolving it with a Gaussian γ; thus we will compute the approximate derivative $\hat{\psi} = \delta' \otimes (\gamma \otimes \phi)$. But convolution is associative, so this is equivalent to $\hat{\psi} = (\delta' \otimes \gamma) \otimes \phi$. The parenthesized expression is the derivative of the Gaussian function, so we see that an approximate derivative of a field can be computed by convolving it with the derivative of a Gaussian (which is easily implemented through neural connectivity):

$$\phi' \approx \gamma' \otimes \phi.$$

The derivative is approximate because of the filter applied to ϕ, the transfer function of which is the Fourier transform of γ, which is itself Gaussian.

It should be noted that such an analysis can be applied when regions A and B are coextensive, and so no real "projection" is involved. For example, A and B might represent two populations of neurons in the same region, so that the connectivity field κ or L reflects how cells of type B depend on neighboring cells of type A. Indeed, A and B might be the *same* cells, if we are describing how their recurrent activity depends on their own preceding activity and that of their neighbors. Thus we might have a linear differential field equation of the form $\dot{\phi}(t) = \kappa \otimes \phi(t)$ or, more generally, $\dot{\phi}(t) = L\phi(t)$. (See Section 4 for examples.)

3.3.4 Convolution over Transformed Coordinates In the definitions of correlation and convolution, Eqs. 1 and 2, the expressions $s - r$ and $r - s$ show us that these operations are sensitive to distance and direction in the domains of the fields, that is, they depend on the coordinates over which the fields are defined. For example, if ψ results from ϕ by a coordinate transformation, $\psi = \phi \circ h^{-1}$, then the results of convolving ψ with a Gaussian γ will not be the same as the results of convolving ϕ with γ. The convolution $\gamma \otimes \phi$ averages over regions that are circular in ϕ's domain, whereas $\gamma \otimes \psi$ averages over circular regions in ψ's domain. For example, because of the logmap transformation between the retina and VI, a Gaussian convolution in VI will not have the effect of a Gaussian convolution in retinal coordinates or vice versa. This sensitivity of convolutions and correlations to the coordinate system can be a problem that needs to be solved or a computational resource that can be exploited.

[3]The Dirac delta is a "generalized function" that has the value zero everywhere except at the origin, where it has the value infinity.

Suppose we have two domains Ω and Ω' such that fields over Ω' are transformations of fields over Ω; let $h : \Omega \longrightarrow \Omega'$ be the coordinate transformation (an isomorphism). For example, Ω and Ω' might be two brain regions (such as the retina and VI), or one or the other might be an external region (such as physical space around the body). Let α and ϕ be two fields over Ω and suppose we want to compute the convolution $\psi = \alpha \otimes \phi$; for example we might want to do a Gaussian convolution in retinal space. However, suppose that the convolution is to be computed by means of fields defined over the transformed domain Ω'. We are given the transformed $\Phi = \phi \circ h^{-1}$ and want to compute Ψ so that $\Psi \circ h = \psi = \alpha \otimes \phi$. We can get this by changing the integration variable of the convolution (assumed to be scalar to keep the example simple):

$$\begin{aligned}
\Psi(u) &= (\alpha \otimes \phi)[h^{-1}(u)], \\
&= \int_{\Omega} \alpha[h^{-1}(u) - x]\phi(x)\mathrm{d}x, \\
&= \int_{\Omega'} \alpha[h^{-1}(u) - h^{-1}(v)]\phi[h^{-1}(v)]\frac{\mathrm{d}v}{h'[h^{-1}(v)]}, \\
&= \int_{\Omega'} \frac{\alpha[h^{-1}(u) - h^{-1}(v)]}{h'[h^{-1}(v)]}\Phi(v)\mathrm{d}v.
\end{aligned}$$

If we define the connectivity field

$$A_{uv} = \frac{\alpha[h^{-1}(u) - h^{-1}(v)]}{h'[h^{-1}(v)]},$$

then the convolution integral becomes

$$\Psi_u = \int_{\Omega'} A_{uv}\Phi_v \mathrm{d}v,$$

which is the integral operator, $\Psi = A\Phi$. This is a linear operator, but not a convolution, which means that it is still implemented by a simple pattern of connectivity, but that it is not a single pattern duplicated throughout the region. (If, as is often the case, the transformation h is a homeomorphism, then it will preserve the topology of Ω, which means that a local convolution α in Ω will translate into local connections A in Ω'.)

We remark without proof that if the domains are of more than one dimension, then the connectivity kernel is defined

$$A_{uv} = \alpha[h^{-1}(u) - h^{-1}(v)]\ J[h^{-1}(v)],$$

where $J[h^{-1}(v)]$ is the Jacobian of h^{-1} evaluated at v.

Now, conversely, suppose we do a convolution $\Psi = \Gamma \otimes \Phi$ in the transformed coordinates; what is its effect in the original coordinates? By a similar derivation we find that $\psi = C\phi$ where the kernel is defined

$$C_{xy} = \Gamma[h(x) - h(y)]\ J[h(y)].$$

In effect, the convolution kernel Γ is projected backward through the transformation h. For example, if, like the logmap transformation, h expands the space in the center of the visual field and compresses it at the periphery, then the back-transformation of Γ will result in a C that defines small receptive fields near the center of the visual field, and large ones near its periphery.

4 Field Dynamics

The field operations considered above are examples of *nonrecurrent* operations, typically implemented by feed-forward connections between neural areas. In this section we will consider *recurrent* operations, which are typically implemented by feed-back or reciprocal connections. Thus there are dynamical relations between several areas that govern the variation in time of one or more fields; these processes are especially important in motor control, since time-varying motor fields in the central and peripheral nervous systems must be generated to control physical movement.

Field dynamics are most conveniently expressed by *differential field equations*, in which the time-derivative $\dot{\psi}(t)$ of a *state field* ψ is given as a function of the current state field $\psi(t)$ and some, possibly time-varying, input field $\phi(t)$:

$$\dot{\psi}(t) = F[\psi(t), \phi(t)].$$

More generally, we may have a system of state fields ψ_k, $k = 1, \ldots, m$, each evolving under the influence of each other and one or more input fields ϕ_k, $k = 1, \ldots, n$. Thus,

$$\dot{\psi}_k(t) = F_k[\psi_1(t), \ldots, \psi_m(t); \phi_1(t), \ldots, \phi_n(t)].$$

(For purposes of mathematical modeling, equations involving second- and higher-order time derivatives can be placed in this form by adding state fields to explicitly represent derivatives, in which case we must carefully distinguish fields represented in neural tissue from those introduced for mathematical convenience.) As before, we may distinguish between the cases in which the dependence is (approximately) linear or not.

4.1 Linear Dynamics

In the (approximately) linear case F can be separated into two linear operators L and M operating on the state and input, respectively; the time derivative of the state is a superposition of the results of these operations:

$$\dot{\psi} = L\psi + M\phi.$$

Next we'll consider several important examples of linear field processes.

A *diffusion process* is defined by a linear differential field equation:

$$\dot{\psi} = k^2 \nabla^2 \psi,$$

where the *Laplacian* is defined:

$$\nabla^2 \psi = \sum_k \frac{\partial^2 \psi}{\partial x_k^2},$$

and the summation is over all the dimensions x_k of the extent of ψ.

Many useful computations can be performed by diffusion processes; for example chemical diffusion processes have been used for finding minimum-length paths through a maze (Steinbeck et al. 1995). Also, diffusion equations have been used to implement Boltzmann machines and simulated annealing algorithms, which have been used to model optimization and constraint-satisfaction problems, such as segmentation and smoothing in early vision, and correspondence problems in stereo vision and motion estimation (Miller et al. 1991, Ting & Iltis 1994).

In the brain, diffusion processes, implemented by the spreading activation of neurons, could be used for planning paths through the environment. For example, a diffusion process is approximated by a network in which each neuron receives activation from its neighbors, without which its activity decays. Thus the change in activity of neuron x is given by

$$\dot{\psi}_x = k^2 \left(-\psi_x + \frac{1}{n} \sum_i \psi_{x_i} \right),$$

where ψ_{x_i} are the activities of its n neighbors x_i. More clearly, writing $\langle \psi_{x_i} \rangle$ for the average activity of its neighbors,

$$\dot{\psi}_x = k^2 (\langle \psi_{x_i} \rangle - \psi_x).$$

The averaging process can be accomplished by convolution with a radial function, such as a Gaussian:

$$\dot{\psi} = k^2 (\gamma \otimes \psi - \psi).$$

Constraints on the path (impassable regions) are represented by neurons whose activity is inhibited; relatively impassable regions can be represented by neurons that are only partly inhibited.

4.2 Nonlinear Dynamics

In the nonlinear case, the variation in the state field ψ is a nonlinear function F of the state and the input field ϕ:

$$\dot{\psi}(t) = F[\psi(t), \phi(t)].$$

Many computational processes, especially optimization processes, can be described by *gradient descent*; this is most commonly seen in low-dimensional vector spaces, but applies as well to field computation, as will now be explained. Often the suitability of a field ψ for some purpose can be measured

by a scalar function $U(\psi)$ (for reasons that will become apparent, we will take *lower* numbers to represent *greater* suitability). For example, ψ might represent an interpretation of sensory data and $U(\psi)$ might represent the internal incoherence of that interpretation (so that the lowest $U(\psi)$ gives the most coherent ψ). More relevantly, ψ might represent a motor plan of some kind, and $U(\psi)$ the difficulty, in some sense, of that plan. Then minimizing $U(\psi)$ gives an optimal plan. By analogy with physical processes, $U(\psi)$ is called a *potential function*.

One way to find a state ψ that minimizes U is by a *gradient-descent process*, that is, a process that causes ψ to follow the gradient $\nabla U(\psi)$ of the potential. The gradient is defined:

$$(\nabla U)_x = \frac{\partial U}{\partial \psi_x}$$

(where, for notational convenience, we treat the field ψ as a high-dimensional vector). The gradient $\nabla U(\psi)$ is a field (over the same domain as ψ) giving the "direction" of change that most rapidly increases U, that is, the relative changes to areas of ψ that will most rapidly increase U. Conversely, the negative gradient $-\nabla U$ gives the direction of change that most rapidly *decreases* U. (This is because ∇ is linear and so $\nabla(-U) = -\nabla U$.)

In a *gradient-descent* process the change of state is proportional to the negative gradient of the state's potential:

$$\dot{\psi} = -r\nabla U(\psi).$$

(The constant r determines the rate at which the process takes place.) The resulting "velocity" field $\dot{\psi}$ is called a *potential flow*.

It is easy to show that a gradient-descent process cannot increase the potential, and indeed it must decrease it unless it is at a (possibly local) minimum (or other saddle point). In this way gradient-descent can be used for optimization (although, in general, we cannot guarantee that a global minimum will be found).

A common, special case occurs when the potential is a quadratic function:

$$U(\psi) = \psi Q \psi + \rho \cdot \psi + \sigma,$$

where by $\psi Q \psi$ we mean the *quadratic form*:

$$\psi Q \psi = \int_\Omega \int_\Omega \psi_x Q_{xy} \psi_y \mathrm{d}y \mathrm{d}x.$$

The *coupling field* Q, which is of higher type than ψ (i.e., Q is a field over $\Omega \times \Omega$), is required to be *symmetric* ($Q_{xy} = Q_{yx}$). In this case the gradient has a very simple (first degree) form:

$$\nabla U(\psi) = 2Q\psi + \rho,$$

where, as usual, $Q\psi$ is the integral operator $(Q\psi)_x = \int_\Omega Q_{xy}\psi_y \mathrm{d}y$. In many cases $\rho = \mathbf{0}$ and gradient descent is a linear process:

$$\dot{\psi} = -rQ\psi.$$

Notice that $-Q_{xy}$ represents the coupling between regions x and y of the state field and therefore how the potential varies with coherence between activity in these parts of the field. If $Q_{xy} > 0$ then the potential will be lower to the extent ψ_x and ψ_y covary (are positive at the same time or negative at the same time) since then $-\psi_x Q_{xy}\psi_y \leq 0$; if $Q_{xy} < 0$, the potential will be lower to the extent they contravary. Thus $-(Q\psi)_x$ gives the change to ψ_x that maximally decreases the potential according to the covariances and contravariances requested by Q.

5 Learning

Representations of motion patterns can be quickly learned and adapted by a variety of field computational methods; many involve the extraction of frequency-domain information from example motions (by application of inner-product or filtering techniques). Invariances in sensorimotor coordination can emerge similarly from simple correlational adaptive algorithms. Since an adequate treatment of field-computational approaches to learning is beyond the scope of this paper, I will give just two examples of the field-computational approach.[4]

5.1 Correlational Learning

Many familiar neural network learning algorithms, including correlational (Hebbian) and back-propagation learning, are easily transferred to the field computation framework. For example, Hebbian learning rules can be described in terms of an *outer product* of fields, $\phi \wedge \psi$:

$$(\phi \wedge \psi)_{xy} = \phi_x \phi_y.$$

(Notice that if ϕ is a field over Ω and ψ is a field over Ω', then $\phi \wedge \psi$ is a field over $\Omega \times \Omega'$.) For example, simple correlational strengthening of an interconnection kernel K resulting from pre- and post-synaptic activity fields ϕ and ψ is given by $\dot{K} = r\ \phi \wedge \psi$, where r is the rate. Such a process might occur through long-term potentiation (LTP).

Recent studies (surveyed in Singer 1995) indicate that moderately weak positive correlations cause synaptic efficacy to be weakened through long-term depression (LTD), while very weak connections have no effect on efficacy. For (biologically realistic) non-negative activity fields, the change in

[4]See Section 6 for a discussion of some representations and MacLennan (1994) for example adaptive algorithms.

the interconnection matrix is given by $\dot{K} = r\upsilon(\phi \wedge \psi)$, where the *upsilon function* is defined:

$$\upsilon(x) = \tanh\ \alpha(x - \theta) - \frac{\tanh\ \beta(x - \eta) + 1}{2}.$$

When $x > \theta$, $\upsilon(x) > 0$ and LTP results, but as x drops below θ, $\upsilon(x)$ becomes negative, achieving its minimum at $x = \eta$; further decreases of x cause $\upsilon(x)$ to approach 0. (The slopes in the LTP and LTD regions are determined by α and β.)

5.2 Gradient Descent

In general, if $F(p_1, \ldots, p_n; \phi_1, \ldots, \phi_n) = F_{\mathbf{p}}(\phi_1, \ldots, \phi_n)$ is some field computational process governed by parameters $p_1, \ldots, p_n$ (such as synaptic weights), and if $M[\phi_1, \ldots, \phi_n; F_{\mathbf{p}}(\phi_1, \ldots, \phi_n)]$ is some performance measure for F on the input fields $\phi_1, \ldots, \phi_n$, then for fixed $\phi_1, \ldots, \phi_n$ we may define a potential field $\mu_{\mathbf{p}} = M[\phi_1, \ldots, \phi_n; F_{\mathbf{p}}(\phi_1, \ldots, \phi_n)]$ over the parameter space. If smaller values of M represent better performance, and if M is bounded below (i.e., there is a best performance), then we can do gradient descent on the parameter space, $\dot{\mathbf{p}} = -r\nabla\mu$.

The same analysis can be applied when F is parameterized by one or more fields (typically, interconnection fields). In this case, gradient descent occurs by gradual modification of the parameter fields. For example, in the case of one parameter field, $\mu_\psi = M[\phi_1, \ldots, \phi_n; F_\psi(\phi_1, \ldots, \phi_n)]$, the descent is given by $\dot{\psi} = -r\nabla\mu$. Of course, more sophisticated hill-descending algorithms can also be implemented by field computation.

6 Examples of Motor Field Computation

6.1 External Force Fields and Motor Basis Fields

Bizzi & Mussa-Ivaldi (1995) survey experiments showing that regions in the spinal chord of the frog define associated force fields in the vicinity of the leg; that it, microstimulation of that spinal region causes the leg to exert a consistent force, which depends on the position of the leg, thus defining a force field over its range of motion. They further show that microstimulation of multiple spinal regions create a force field that is the linear superposition (sum) of the individual force fields, and that this superposition determines the location to which the leg moves. Specifically, a time-varying force field $\mathbf{F}(t)$ results from a linear superposition of time-varying *basis fields* $\phi_k(t)$, each generated by a premotor circuit in the frog's spinal chord:

$$\mathbf{F}(t) = \sum_k c_k \phi_k(t).$$

As few as four convergent force fields ϕ_k are sufficient to generate a wide variety of resultant fields.

6.2 Population Coding of Direction

Georgopoulos (1995) surveys research on population coding in motor cortex of the direction of arm motion. The population codes are naturally treated as fields, and the transformations of directions are simple field computations. We consider a region Ω in motor cortex in which activity is observed in anticipation of reaching motions. Each cell $u \in \Omega$ has a preferred direction $\mathbf{D}_u$ in three-dimensional space. Cell activity ϕ_u falls off with the cosine of the angle θ_u between the reaching direction $\mathbf{r}$ and the preferred direction $\mathbf{D}_u$. Since (for normalized vectors) the cosine is equal to the inner product of the vectors, $\mathbf{r} \cdot \mathbf{D}_u = \cos\theta_u$, we can express the activity:

$$\phi_u = a + b\mathbf{r} \cdot \mathbf{D}_u, \tag{3}$$

for some constants a and b.[5] Thus the motor cortex represents a vector field $\mathbf{D}$ of the preferred directions, and the population coding of an intended motion $\mathbf{r}$ is a scalar activity field $\mathbf{r} \cdot \mathbf{D}$ given by the inner product of the motion with the preferred-direction field.

There is another way of looking at the population coding ϕ of a motion $\mathbf{r}$, which is sometimes more illuminating. Since all the neurons have the same receptive field profile, we may rewrite Eq. 3 in terms of a radial function ϱ of the difference between the preferred and intended direction vectors:

$$\phi_u = \varrho(\mathbf{D}_u - \mathbf{r}),$$

where

$$\varrho(\mathbf{v}) = a + b - b\|\mathbf{v}\|^2/2.$$

This is because the Euclidean distance is related to the inner product in a simple way:

$$\begin{aligned} a + b - b\|\mathbf{D}_u - \mathbf{r}\|^2/2 &= a + b - b(\|\mathbf{D}_u\|^2 + \|\mathbf{r}\|^2 - 2\mathbf{D}_u \cdot \mathbf{r})/2 \\ &= a + b\mathbf{D}_u \cdot \mathbf{r} \\ &= a + b\cos\theta_u \end{aligned}$$

(provided $\|\mathbf{r}\| = 1 = \|\mathbf{D}_u\|$).

Now let ψ be the direction field, defined over three-dimensional space, that corresponds to ϕ. That is, the value of ϕ at neural location u equals the value of ψ at spatial location $\mathbf{D}_u$, or $\phi_u = \psi_{\mathbf{D}_u}$, which we may abbreviate $\phi = \psi \circ \mathbf{D}$. For simplicity we suppose $\mathbf{D}$ is one-to-one, so we can define ψ by $\psi = \phi \circ \mathbf{D}^{-1}$. Notice that $\mathbf{D}$ effects a change of coordinates from neural coordinates to three-dimensional space. The direction field ψ can also be expressed as the result of convolving the receptive field ϱ with an idealized direction field $\delta_{\mathbf{r}}$, a Dirac delta, which has an infinite spike at $\mathbf{r}$ but is zero elsewhere:

$$\psi = \varrho \otimes \delta_{\mathbf{r}}.$$

[5] For a typical case shown in Georgopoulos (1995, Fig. 32.1) and normalized vectors, it appears $a \approx 30$ impulses/sec. and $b \approx 20$ impulses/sec.

This is because convolving ϱ with $\delta_{\mathbf{r}}$ effectively translates the center of ϱ to $\mathbf{r}$; equivalently, the convolution blurs the idealized direction field $\delta_{\mathbf{r}}$ by the receptive field profile ϱ.

6.3 Continuous Transformation of Direction Fields

There is considerable evidence that humans and monkeys are able to continuously transform images for various purposes. Aside from introspection, such evidence comes from the behavioral experiments pioneered by Shepard (e.g. Shepard & Cooper 1982) and, more recently, from direct neuronal measurement of motor cortex (surveyed in Georgopoulos 1995).

Droulez & Berthoz (1991b) give an algorithm for the continuous transformation of direction fields, specifically, for the updating, when the eye moves, of the remembered location, relative to the retina, of an ocular saccade.[6] Suppose the field ϕ is a population code in retinal coordinates for the destination of the saccade. If in time Δt the eye moves by a vector $\Delta\mathbf{r}$ in retinal coordinates, then the field encoding the destination of the saccade must be updated according to the equation

$$\phi(\mathbf{r} + \Delta\mathbf{r}, t + \Delta t) = \phi(\mathbf{r}, t).$$

Eye motion is assumed to be encoded by a two-dimensional rate-encoded velocity vector $\mathbf{v}$, which gives the eye velocity in retinal coordinates. It is easy to show that

$$\phi(t + \Delta t) = \phi(t) + \Delta t \ \mathbf{v} \cdot \nabla\phi(t). \tag{4}$$

(The gradient $\nabla\phi$ points in the direction of the peak, provided there is only one peak; if there are multiple targets, it points to the nearest target.) This equation, which gives a discrete update after a time Δt, can be converted into a equation for the continuous updating of ϕ by taking the limit as $\Delta t \longrightarrow 0$:

$$\dot{\phi} = \mathbf{v} \cdot \nabla\phi.$$

This can be understood as follows: Since $\mathbf{v}$ represents the motion of the eye relative to the retinal field, $-\mathbf{v}$ represents the direction in which the field peak should move. In front the peak (that is, in its direction of required movement), the gradient, which points toward the peak, points in the opposite direction to $-\mathbf{v}$. Therefore $-\mathbf{v} \cdot \nabla\phi$ at that point will be negative, which means that $\dot{\phi} = \mathbf{v} \cdot \nabla\phi > 0$, and the field intensity in the front of the peak increases. Conversely, behind the peak the gradient points in the same direction as the required movement, so $-\mathbf{v} \cdot \nabla\phi > 0$, which means $\dot{\phi} = \mathbf{v} \cdot \nabla\phi < 0$, and the field intensity on the back of the peak decreases. Therefore, the peak moves in the required direction.

[6] This process may take place in the superior colliculus, frontal eye field or posterior parietal cortex (Droulez & Berthoz 1991b).

Equation 4 must be recast for neural computation, since the vector field $\nabla\phi$ has to be represented by two neural populations (for the two dimensions of retinal coordinates). Thus we write

$$\mathbf{v} \cdot \nabla\phi = v_{\mathrm{x}} \frac{\partial\phi}{\partial x} + v_{\mathrm{y}} \frac{\partial\phi}{\partial y}.$$

Since the neural population is discrete and the neurons have receptive fields with some diameter, the neural representation imposes a low-pass filter on the direction field. Writing γ_{xy} for a two-dimensional Gaussian, the filtered field can be written $\gamma_{\mathrm{xy}} \otimes \phi$ and substituted into Eq. 4:

$$\begin{aligned} \phi(t+\Delta t) &= \gamma_{\mathrm{xy}} \otimes \phi + \Delta t \; \mathbf{v} \cdot \nabla(\gamma_{\mathrm{xy}} \otimes \phi) \\ &= \gamma_{\mathrm{xy}} \otimes \phi + \Delta t \left[v_{\mathrm{x}} \frac{\partial(\gamma_{\mathrm{xy}} \otimes \phi)}{\partial x} + \frac{\partial(\gamma_{\mathrm{xy}} \otimes \phi)}{\partial y} \right] \end{aligned}$$

As we've seen, the derivatives of the filtered field can be written as convolutions with derivatives of Gaussians, so $\partial(\gamma_{\mathrm{xy}} \otimes \phi)/\partial x = \gamma'_{\mathrm{x}} \otimes \phi$, where γ'_{x} is a derivative of a Gaussian along the x-axis and constant along the y-axis. Thus,

$$\phi(t+\Delta t) = \gamma_{\mathrm{xy}} \otimes \phi + \Delta t (v_{\mathrm{x}} \gamma'_{\mathrm{x}} \otimes \phi + v_{\mathrm{y}} \gamma'_{\mathrm{y}} \otimes \phi).$$

Significantly, when Droulez & Berthoz (1991b) started with a one-dimensional network of the form

$$\alpha \otimes \phi + \Delta t \; v\beta \otimes \phi$$

and trained it, by a modified Hebbian rule, to compute the updated population code, they found that after training α was approximately Gaussian, and β was an approximate derivative of a Gaussian.

Droulez & Berthoz (1991a) suggest biologically plausible neural circuits that can update the direction field ϕ, which can be expressed in field computational terms as follows. A field of interneurons S (sum) forms the sum of the activities of nearby neurons, $S = \gamma_{\mathrm{xy}} \otimes \phi$, while interneuron fields G_{x} and G_{y} estimate the partial derivatives by a means of excitatory and inhibitory synapses, $G_{\mathrm{x}} = \gamma'_{\mathrm{x}} \otimes \phi$, $G_{\mathrm{y}} = \gamma'_{\mathrm{y}} \otimes \phi$. Next, a field of interneurons P (product) computes the inner product of the velocity vector and the field gradient by means of conjunctive synapses: $P = v_{\mathrm{x}} G_{\mathrm{x}} + v_{\mathrm{y}} G_{\mathrm{y}}$. The neurons in the direction field compute the sum of the S and P interneurons, which then becomes the new value of the direction field, $\phi = S + P$. Thus Droulez & Berthoz's (1991a) proposed neuronal architecture corresponds to the following field equations, all implemented through local connections:

$$\begin{aligned} S &= \gamma_{\mathrm{xy}} \otimes \phi, \\ G_{\mathrm{x}} &= \gamma'_{\mathrm{x}} \otimes \phi, \\ G_{\mathrm{y}} &= \gamma'_{\mathrm{y}} \otimes \phi, \\ P &= v_{\mathrm{x}} G_{\mathrm{x}} + v_{\mathrm{y}} G_{\mathrm{y}}, \\ \phi &= S + P \end{aligned}$$

6.4 Fields Associated with Posterior Parietal Cortex

Andersen (1995) surveys research indicating that the transformation from retina-centered coordinates to head- or body-centered coordinates can be understood in terms of fields associated with neurons in area 7a of the posterior parietal cortex. When the eye position is fixed, these neurons exhibit an ordinary receptive field (defined over retinal coordinates) in their response to a stimulus. On the other hand, when the position of the stimulus on the retina is fixed, then these neurons exhibit a response that varies linearly with eye position; this is described by a *linear gain field*, defined over eye position, and has a characteristic direction. Specifically, a linear gain field λ is described by a direction vector $\mathbf{d}$, which is its gradient, $\mathbf{d} = \nabla\lambda$; thus, $\lambda_{\mathbf{p}} = \mathbf{d} \cdot \mathbf{p}$ at all positions $\mathbf{p}$. Under normal conditions the response of the neuron is a product of the receptive field and the linear gain field, and so its response is defined over the four dimensions of retinal and eye position. The result is a neuron tuned to particular locations in *head-centered* space, but only for certain ranges of eye position. Therefore, single neurons cannot encode locations in head-centered space, but a field of neurons can combine their responses into a population code for head-centered locations. The resulting field has a well-defined minimum in head-centered space, which can represent the destination of a motion (such as a saccade) and, by means of its gradient, a path to that destination.

Andersen (1995) also surveys studies of ocular motion planning in the lateral intraparietal area of the posterior parietal cortex (see also Goodman & Andersen 1989). Microstimulation of neurons create eye movements that can be described as vector fields (giving the direction and amount of motion) over head-centered coordinates. Three kinds of fields $\mathbf{V}$ are typically found: (1) constant vector fields ($\mathbf{V}_{\mathbf{p}} = \mathbf{v}$ for all locations $\mathbf{p}$), (2) vector fields of constant direction but decreasing amplitude ($\mathbf{V}_{\mathbf{p}} = (\mathbf{v}^{\mathrm{T}}\mathbf{p}\mathbf{v})^{+}$, that is, the positive part of $\mathbf{v}^{\mathrm{T}}\mathbf{p}\mathbf{v}$), and (3) weakly convergent vector fields, which rarely reverse direction. On the other hand, in simulation studies, microstimulation of two or more neurons created strongly convergent motion fields by vector summation of the individual fields of the neurons. The gradient of such s field defines the paths, in head-centered space, to the location defined by the minimum.

6.5 Probability Density Functions

Sanger (submitted) argues for the interpretation of neural activity in terms of external (sensory or motor) conditional probability density functions (CPDFs) corresponding to their generalized receptive fields. Thus, a neuron i has an associated CPDF σ_i defined over some bounded range Ω of external phenomena. In particular, the firing of neuron i represents phenomenon $x \in \Omega$ with conditional probability $\sigma_i(x)$. Clearly, such a CPDF is a field, and so we can say that each neuron has an associated *conditional probability field*. The conditional probability field associated with a population

of neurons can then be defined in terms of field operations on the fields of the constituent neurons. For example, Sanger shows that over small time intervals (such that spiking is relatively unlikely), the field of the population is a product of the fields of the neurons that spike in that interval:

$$\sigma_{\text{pop}} = \prod_{i \in \text{spike}} \sigma_i,$$

where $\prod$ represents a pointwise product of the fields, $\sigma_{\text{pop}}(x) = \prod_{i \in \text{spike}} \sigma_i(x)$. Further, Sanger shows that for any smooth mapping $y = f(x)$, there is a corresponding *piecewise linear* mapping on the probability fields P_y and P_x, which is given by an integral operator, $P_y = KP_x$.

7 Representation of Motion

7.1 Introduction

There are several ways that motion can be represented in fields and generated through field computation. Each has advantages and disadvantages in terms of efficiency of representation, flexibility and other factors.

7.2 Direct (Spatial) Representation

One of the simplest ways to represent a trajectory $\phi(t)$ is by direct spatial encoding of the time dimension; then the trajectory can be read sequentially from the fixed field. (This process is like playing an audio tape.) More precisely, suppose $\phi_u(t)$ is a time-varying field defined over an extent Ω (that is, $u \in \Omega$), and we want to generate it over the relative time interval $t \in [0, T]$. Let $h : [0, T] \longrightarrow \Omega'$ be a mapping from the time interval to another domain of spatial extension; then the trajectory $\phi_s(t)$ is encoded by a fixed field ψ_{uv} over $\Omega \times \Omega'$ defined by:

$$\psi[u, h(t)] = \phi_u(t).$$

The field ψ_{uv} is "read out" by sweeping v from $h(0)$ to $h(T)$.

Since the area of the field ψ is proportional to the duration of the signal ϕ, such a representation is feasible only for signals that are comparatively smooth with respect to their duration. (Specifically, by the Nyquist theorem, there must be as least two representational units v per unit time for the highest frequency component of ϕ.)

7.3 Frequency-domain Representation

Frequency encoding generates a signal ψ from its (discrete or continuous) Fourier transform Ψ, which is represented spatially. Suppose we have a

signal $\psi_u(t)$ of duration T (or periodic with period T); write it as a discrete Fourier series:

$$\psi_u(t) = \sum_{k=0}^{n} \alpha_{ku} \cos\left(\frac{2\pi kt}{T} - \phi_{ku}\right). \tag{5}$$

(The number of coefficients n is determined by the Nyquist frequency: twice the highest frequency in ψ.)

The signal then is determined by the amplitude fields $\alpha_1, \ldots, \alpha_n$ and the phase fields $\phi_1, \ldots, \phi_n$ (together they constitute the discrete Fourier transform Ψ). The signal is generated by using them to control the amplitude and phase of a "bank" of sinusoidal signal generators, in accord with Eq. 5. (Of course, it's not essential that the signal generators be sinusoidal, since the Fourier expansion can be done in terms of any orthonormal basis.)

The approach is easily extended to the continuous Fourier transform; write

$$\psi_u(t) = \frac{1}{2\pi} \int_{-\omega_{\max}}^{\omega_{\max}} \Psi_{u\omega} e^{-i\omega t} \mathrm{d}\omega.$$

Now define a one-dimensional field of signal generators, $\varepsilon_\omega(t) = e^{-i\omega t}/2\pi$, implemented, perhaps, by pairs of neurons in quadrature phase; then the signal is constructed by

$$\psi_u(t) = \int_{-\omega_{\max}}^{\omega_{\max}} \Psi_{u\omega} \varepsilon_\omega(t) \mathrm{d}\omega = \Psi_u \varepsilon(t),$$

which we may abbreviate $\psi = \Psi\varepsilon$.

The Fourier representation is especially appropriate when frequency-domain transformations need to be applied to the signal, or when the signal is periodic (since only one cycle needs to be encoded). If the Fourier representation is translated by $\Delta\omega$ along the frequency axis, then the overall duration of one cycle changes $T \longrightarrow T/(1 + \Delta\omega/\omega)$ (so an increase of frequency leads to a decrease of duration and vice versa). Conversely, the duration of the signal cannot be changed without changing its frequency content (since the fundamental frequency is the reciprocal of the duration).

7.4 Gabor Representation

We have seen that a field can represent a trajectory in either the time domain or the frequency domain. Since each has its advantages and disadvantages, often a combined representation is more suitable. In such a representation we have a time-varying spectrum.

The foundation for such a representation was laid fifty years ago by Dennis Gabor, who also received the Nobel Prize for his invention of holography. Gabor (1946) observed that we perceive sound in terms of amplitude and pitch simultaneously, that is, auditory perception is not entirely in the time domain or the frequency domain. He showed that any signal of finite duration and bandwidth could be decomposed into a finite number of elementary

information units, which he called *logons*. Each such unit controls the amplitude and phase of a *Gabor elementary function*, which is an elementary signal localized in time and frequency. The relevance of this to motor control is that any motor control signal has a calculable Gabor-information content,[7] which determines a finite number of coefficients necessary and sufficient to generate that signal. Pribram et al. (1984) and Pribram (1991, 139–144) provide evidence for Gabor elementary functions in motor control.

More precisely, at time t the measurement of a frequency component f in a signal will require that the signal be sampled for some finite duration Δt. Further, the uncertainty Δf in the measured frequency will be less the longer the signal is sampled. Indeed, Gabor proves $\Delta t \Delta f \geq 1/4\pi$ (the so-called Gabor Uncertainty Principle).[8] (An intuitive presentation of the proof can be found in MacLennan 1991.) Therefore $\Delta t \Delta f = 1/4\pi$ defines the maximum possible definition of a (finite duration, finite bandwidth) signal. A signal of duration T and bandwidth F can be divided into a finite number of elementary "information cells" of duration Δt and bandwidth Δf, each localized at a different time and frequency. Each cell has an associated complex coefficient, which gives the phase and amplitude of the signal in the corresponding cell. Let $M = T/\Delta t$ and $N = F/\Delta f$; then there are MN elementary information cells; in Gabor's terms, the signal represents MN logons of information, namely, the MN coefficients associated with the cells.[9] This is the most information that can be represented by the signal, and these MN complex coefficients are sufficient to regenerate the signal (which is its relevance for motor control).

Let the cells be labeled (j, k) for $j = 0, \ldots, M-1$ and $k = 0 \ldots, N-1$. Then cell (j, k) is centered at time $j\ \Delta t$ and frequency $k\ \Delta f$. Each cell corresponds to a *Gabor elementary function* localized to that time and frequency, one form of which is a Gaussian-modulated sinusoid:

$$G_{jk}(t, \phi) = \exp\left[\frac{-\pi(t\ -\ j\Delta t)^2}{\beta^2}\right] \cos[2\pi k \Delta f(t\ -\ j\Delta t\ -\ \phi)],$$

where $\beta = 2\sqrt{\pi}\ \Delta t = \Delta f\ /\ 2\sqrt{\pi}$ (the standard deviation of the Gaussian is Δt). A signal $\psi(t)$ is then a superposition of these elementary functions with amplitudes α_{jk} and phase delays ϕ_{jk}:

$$\psi(t) = \sum_{j=0}^{M-1} \sum_{k=0}^{N-1} \alpha_{jk} G_{jk}(t, \phi_{jk}).$$

The coefficients α_{jk} and ϕ_{jk} are determined uniquely by the signal ψ.

[7] Gabor's notion of information is not the same as Shannon's; they are complementary rather than mutually exclusive. See MacLennan (1991) and citations therein for a discussion.

[8] The precise constant, $1/4\pi$ in this case, depends on the quantification of the uncertainty of measurement (MacLennan 1991).

[9] For technical reasons (see MacLennan 1991), these MN complex coefficients comprise only $2MN - M$, as opposed to $2MN$, independent real coefficients.

The Gabor representation shows us how a signal can be generated from the control coefficients α_{jk} and ϕ_{jk}: during the jth time interval of length Δt we use the coefficients to control a bank of Gaussian-modulated sinusoid generators (at frequencies $k\ \Delta f$); α_{jk} controls the amplitude of generator k and ϕ_{jk} controls its phase.[10]

Although the clocking out at discrete time intervals of the coefficients is not impossible, it may seem a little unnatural. This can be avoided by replacing the discrete matrices (α_{jk}) and (ϕ_{jk}) by continuous fields. In this approach the Gabor elementary function generators operate on a continuum of frequencies in the signal's bandwidth:

$$G_{\tau\nu}(t,\phi) = \exp\left[\frac{-\pi(t-\tau)^2}{\beta^2}\right]\cos[2\pi\nu(t-\tau-\phi)],$$

The output signal is then generated by an integration:

$$\psi(t) = \int_0^T \int_0^F \alpha_{\nu\tau} G_{\tau\nu}(t,\phi_{\nu\tau})\mathrm{d}\nu\mathrm{d}\tau.$$

In fact, the output can be generated by a temporal convolution of the control fields and a bank of Gabor signal generators, but the details will not be presented here. It might be objected that the control fields α and ϕ would occupy more neural space than either a direct or Fourier representation, but the control fields are relatively low resolution and may be represented more compactly. The inequality $\Delta t\ \Delta f \geq 1/4\pi$ gives the tradeoff in required resolution between the time and frequency axes of the control fields.

Unlike the Fourier representation, the Gabor representation allows frequency content and rate to be controlled independently. Thus the amplitude and phase fields (α,ϕ) can be "clocked out" at a different rate from that at which they were stored, or even at a varying rate, without affecting the moment to moment frequency content of the signal. Conversely, shifting the representing fields (α,ϕ) along the frequency axis shifts the frequency content of the signal, but does not affect its duration or the time-evolution of its spectrum. That is, the rate or time-evolution of the signal can be controlled independently of the frequency band in which it is expressed.

7.5 Wavelet & Multiresolution Representations

The Gabor representation uses the same temporal resolution Δt in each frequency band f_k. However, a Δt that is a good resolution at a low frequency may not be a good resolution at a high frequency. Therefore, in a *multiresolution representation* higher frequency bands may have a smaller (finer) Δt than lower frequency bands. Of course, the Gabor relationship

[10] There is an additional complication in that the Gaussian envelopes extend outside the nominal Δt (= standard deviation) widths of the elementary function. This could be solved by two or three banks of generators activated in rotation; however a better solution lies in the Gabor transform, discussed below.

$\Delta t\ \Delta f \geq 1/4\pi$ still holds, so the frequency resolution Δf must increase (i.e. become coarser) at higher frequencies. This is often acceptable, however, since the ratio of Δf to the frequency remains constant (so this is also called a "constant Q" representation, since $Q = \Delta f\ /\ f$).

In the most common arrangement, the central frequencies of the frequency bands increase by powers of 2, $f_k = 2^k f_0$. Therefore, the widths of the frequency bands also increase by powers of 2, $\Delta f_k = 2^k \Delta f_0$, but the time resolutions decrease (become finer) by powers of 2, $\Delta t_k = 2^{-k} \Delta t_0$. In this case the elementary functions are generated by contracting and translating a single *mother wavelet*:

$$W_{jk}(t) = W_{00}[2^k(t - j\ \Delta t_0)],$$

for $j = 0,\ \ldots,\ 2^k T/\Delta t_0$ and $k = 1, \ldots, N$. The Gabor elementary function, or a slight variant of it called the Morlet wavelet, can be used as a mother wavelet. The signal then is represented by a linear superposition of wavelets:

$$\psi(t) \quad = \quad \sum_{k=0}^{N} \sum_{j=0}^{2^k T\ /\ \Delta t_0} c_{jk} W_{jk}(t).$$

The generation of the signal is controlled by the triangular array of coefficients c_{jk}. Like the continuous Gabor transform, there is also a continuous wavelet transform that represents the coefficients in a continuous field. Also like the Gabor transform, the wavelet transform allows independent control of frequency content and time-evolution. However, because of the essentially exponential measurement of frequency (2^k in the wavelet vs. k in the Gabor), translation along the frequency axis causes dilation or compression of the signal's spectrum. A shift of Δf changes the instantaneous spectrum from $\Psi(f)$ to $\Psi(f/2^{\Delta f})$. Much more could be said about the information processing affordances of these representations, but it is beyond the scope of this paper.

7.6 Constraint satisfaction

7.6.1 Representation as potential field Many problems in motor control involve the satisfaction of constraints; in some cases the satisfaction is inherent in the mechanics of the motor system (and satisfaction takes place through execution of the motion), but in others, such as path planning, the optimum is determined before motion begins and may need to be revised as exigencies arise during its execution.

As already discussed (Sections 2.4 and 4.2), constraints on motion are represented conveniently by a potential field over a spatial map. The potential field representation is quite general. For example, in addition to the representation of hard constraints, increased potential can represent the relative difficulty of motion through a region of space. In this way, a path can be chosen that minimizes "work" (as defined by the potential function).

Further, the potential can be defined over abstract spaces; for example, planning a path through a "lexical space" could be a part of sentence generation. We will consider several ways in which an optimal path can be found by field computation.

7.6.2 Least Action Principles There are many physical "least action principles," in which local behavior (of a particle in a field, for example) causes the minimization of some global measure of "action" (e.g., time, distance, energy dissipation, entropy generation).[11] These processes are often governed by fields, and therefore some optimization and constraint-satisfaction processes in the brain may be implemented through corresponding field computations.[12] One example will be discussed briefly.

In the same way that electromagnetic radiation (such as light) "sniffs out" in parallel a minimum-time path through space (Fermat's Principle), so also neural impulse trains can find a minimum-time path through a neural network. If transmission delays encode the difficulty of passage through a region of some (concrete or abstract) space, then the pulse train will follow the path of least difficulty, and it will automatically shift in parallel to a new optimum if regions change in difficulty; it is not necessary to reinitiate the path planning process from the beginning.

This works because, near an optimum path, the cost does not vary, to a first approximation, with small perturbations of the path, thus the impulses passing near to the optimal path tend to stay in phase. On the other hand, farther away from the optimum the cost does vary, to a first approximation, with small perturbations, so impulses on nearby paths tend to differ in phase. As a result the signals along nonoptimal paths tend to cancel each other out, so only the signals along near-optimal paths have significant amplitude.[13] When difficulties change, the signals near the new optimum tend to reinforce each other, while those that are no longer near an optimum begin to cancel each other out.

Suppose the constant c represents the encoding of difficulty in terms of time delay (in units of difficulty per millisecond, for example), so a time difference of Δt represents a difficulty difference of $c\Delta t$. If the impulses have period T, then we can see that for $\Delta t \gg T$, signals will tend to cancel, whereas for $\Delta t \ll T$ they will tend to reinforce. Thus, impulses of period T will be sensitive to differences in difficulty much greater than cT and insensitive to those much less than cT; they will find paths within cT of the optimum. The sensitivity of search process can be adjusted by varying the

[11] For a clear, insightful introduction to least action principles, it is difficult to do better than Feynman et al. (1963–5, ch. II.19).

[12] For example, least action principles are fundamental to Pribram's (1991) *holonomic brain theory* (see especially Apps. A, B).

[13] For this cancelation to occur, the impulses must be shaped so that their average amplitude is zero. Also, the neurons must sample sufficiently many paths coming into their region to ensure that cancelation is possible; in effect, the neural net must represent the search space at sufficiently high resolution.

impulse frequency (higher frequency for a tighter optimum). Specifically, if the paths converging on a neuron represent a range of difficulties of at least cT, then the neuron will be inactive, showing that it's not near the optimal path. The neuron becomes more active, reflecting its nearness to the optimum, as the range of input difficulties decreases below cT.

Further, the amplitude of the impulses can be used to encode the confidence in the difficulty estimate: regions of the space for which this confidence is low will transmit signals more weakly than high-confidence regions. In this way, difficulty estimates are weighted by their confidence. Specifically, the effect on the signal of passing through a region of space is represented by multiplying by a complex number $ke^{id/c}$, where d is the difficulty estimate and k is the confidence of that estimate. Such a complex multiplication could be accomplished by synaptodendritic transmission, which introduces both an amplitude shift k (reflecting confidence) and a time delay d/c (representing difficulty). Such amplitude/phase modulations would be relatively fixed, subject to slow adaptive mechanisms. However, the same can be accomplished more dynamically (allowing, for instance, an environmental potential field to be loaded into a brain region) by using an external bias to control the phase shift dynamically (Hopfield 1995) and a signal to a conjunctive synapse to control the amplitude dynamically.

7.6.3 Multiresolution satisfaction of constraints Constraints can also be satisfied by gradient descent on a potential surface representing their lack of satisfaction (Sects. 4.2 and 5.2). However, a problem with relaxation techniques is that they may get trapped in local minima. One way to avoid this is to do the relaxation on a multiresolution representation of the potential function. At low resolution, local minima will tend to be averaged away, so relaxation on a low-resolution representation will tend to move toward the global minimum. By gradually increasing the resolution, the system can be allowed to settle into a more accurate representation of the global minimum. This can be accomplished, for example, by gradually activating the higher frequency bands of a Gabor or wavelet representation of the potential surface.

8 Concluding Remarks

We have seen that field computation deals with information processing in terms of fields, which may be described as continuous distributions of data. Many neural phenomena are conveniently described as fields, including neuron activity from large (brain area) to small (dendritic) scales, and it is often useful to describe motor control and sensorimotor coordination in terms of external fields such as force fields and sensory images. We have surveyed the basic concepts of field computation, including both feed-forward field operations and field dynamics resulting from recurrent connections. Adaptive and learning mechanisms were discussed briefly. The application of

field computation to motor control was illustrated by several examples: external force fields associated with spinal neurons, population coding of direction in motor cortex, continuous transformation of direction fields, and linear gain fields and coordinate transformations in posterior parietal cortex. Next we surveyed some field-based representations of motion, including direct, Fourier, Gabor and wavelet or multiresolution representations. Finally we considered briefly the application of these representations to constraint satisfaction, which has many applications in motor control.

References

Andersen, R. A. (1995). Coordinate transformations and motor planning in posterior parietal cortex, *in* M. S. Gazzaniga (ed.), *The Cognitive Neurosciences*, MIT Press, pp. 519–532.

Baron, R. J. (1987). *The Cerebral Computer: An Introduction to the Computational Structure of the Human Brain*, Lawrence Erlbaum.

Bizzi, E. & Mussa-Ivaldi, F. A. (1995). Toward a neurobiology of coordinate transformation, *in* M. S. Gazzaniga (ed.), *The Cognitive Neurosciences*, MIT Press, pp. 495–506.

Changeux, J.-P. (1985). *Neuronal Man: The Biology of Mind*, Oxford University Press. Transl. by L. Garey.

Droulez, J. & Berthoz, A. (1991a). The concept of dynamic memory in sensorimotor control, *in* D. R. Humphrey & H.-J. Freund (eds), *Motor Control: Concepts and Issues*, John Wiley & Sons, pp. 137–161.

Droulez, J. & Berthoz, A. (1991b). A neural network model of sensoritopic maps with predictive short-term memory properties, *Proceedings of the National Academy of Science USA* **88**: 9653–9657.

Ferster, D. & Spruston, N. (1995). Cracking the neuronal code, *Science* **270**: 756–757.

Feynman, R. P., Leighton, R. B. & Sands, M. (1963–5). *The Feynman Lectures on Physics*, Addison-Wesley.

Gabor, D. (1946). Theory of communication, *Journal of the Institution of Electrical Engineers* **93 (III)**: 429–457.

Georgopoulos, A. P. (1995). Motor cortex and cognitive processing, *in* M. S. Gazzaniga (ed.), *The Cognitive Neurosciences*, MIT Press, pp. 507–517.

Goodman, S. J. & Andersen, R. A. (1989). Microstimulation of a neural-network model for visually guided saccades, *Journal of Cognitive Neuroscience* **1**: 317–326.

Hogan, N. (1984). Organizing principle for a class of voluntary movements, *Journal of Neuroscience* **4**: 2745–2754.

Hopfield, J. J. (1995). Pattern recognition computation using action potential timing for stimulus representation, *Nature* **376**: 33–36.

Khatib, O. (1986). Real-time obstacle avoidance for manipulators and mobile robots, *International Journal of Robotics Research* **5**: 90–99.

Knudsen, E. J., du Lac, S. & Esterly, S. D. (1987). Computational maps in the brain, *Annual Review of Neuroscience* **10**: 41–65.

MacLennan, B. J. (1990). Field computation: A theoretical framework for massively parallel analog computation, parts I–IV, *Technical Report CS-90-100*, Computer Science Department, University of Tennessee, Knoxville. Accessible via URL `http://www.cs.utk.edu/~mclennan`.

MacLennan, B. J. (1991). Gabor representations of spatiotemporal visual images, *Technical Report CS-91-144*, Computer Science Department, University of Tennessee, Knoxville. Accessible via URL `http://www.cs.utk.edu/~mclennan`.

MacLennan, B. J. (1993). Information processing in the dendritic net, *in* K. H. Pribram (ed.), *Rethinking Neural Networks: Quantum Fields and Biological Data*, Lawrence Erlbaum.

MacLennan, B. J. (1994). Continuous computation and the emergence of the discrete, *in* K. H. Pribram (ed.), *Origins: Brain & Self-Organization*, Lawrence Erlbaum.

Miller, M. I., Roysam, B., Smith, K. R. & O'Sullivan, J. A. (1991). Representing and computing regular languages on massively parallel networks, *IEEE Transactions on Neural Networks* **2**: 56–72.

Pribram, K. H. (1991). *Brain and Perception: Holonomy and Structure in Figural Processing*, Lawrence Erlbaum.

Pribram, K. H., Sharafat, A. & Beekman, G. J. (1984). Frequency encoding in motor systems, *in* H. T. A. Whiting (ed.), *Human Motor Actions — Bernstein Reassessed*, Elsevier, pp. 121–156.

Rimon, E. & Koditschek, D. E. (1989). The construction of analytic diffeomorphisms for exact robot navigation on star worlds, *Proceedings of the 1989 IEEE International Conference on Robotics and Automation, Scottsdale, Ariz.*, IEEE Publishing Service, New York, pp. 21–26.

Sanger, T. D. (submitted). Probability density estimates for the interpretation of neural population codes, *Journal of Neurophysiology* .

Schwartz, E. L. (1977). Spatial mapping in the primate sensory projection: Analytic structures and relevance to perception, *Biological Cybernetics* **25**: 181–194.

Shepard, R. N. & Cooper, L. (1982). *Mental Images and their Transformations*, MIT Press.

Singer, W. (1995). Development and plasticity of cortical processing architectures, *Science* **270**: 758–764.

Steinbeck, O., Tóth, A. & Showalter, K. (1995). Navigating complex labyrinths: Optimal paths from chemical waves, *Science* **267**: 868–871.

Ting, P.-Y. & Iltis, R. A. (1994). Diffusion network architectures for implementation of Gibbs samplers with applications to assignment problems, *IEEE Transactions on Neural Networks* **5**: 622–638.