Computational Cognitive Neuroscience: Building and Testing Biologically Plausible Computational Models of Neuroscience, Neuroimaging, and Behavioral Data

F. Gregory Ashby
Vivian V. Valentin

2007

The cognitive neuroscience revolution has profoundly altered the nature of theories in cognitive psychology. For many years, the only charge of these theories was to account for purely behavioral data from cognitive experiments that typically were performed on healthy young adults. Now, however, the validity of a cognitive theory may also be challenged by data from a wide variety of other sources—including functional magnetic resonance imaging (fMRI), neuropsychological patient studies, recordings of event-related potentials (ERPs), transcranial magnetic stimulation studies, and single-unit recordings. Clearly the converging evidence provided by these many methods adds tremendous new constraints to the underlying theories, and thereby almost guarantees faster progress in our understanding of the behaviors of interest. But the huge variety of data sources that a successful theory must resolve also greatly increases the difficulty of theory construction. In fact, a new type of theory is required, and along with it, new methods for constructing those theories.

This chapter describes a method we have developed over the past 5 or 6 years for constructing neurobiologically plausible computational models of behavior. The method is quite general and can be applied to a wide variety of behavioral phenomena. The components in the models that are constructed with this method are units that
correspond to groups of similar cells in specific brain regions (e.g., cortical columns or hypercolumns). The resulting models are also quite flexible with respect to the types of data they can be tested against. For example, we describe detailed methods for fitting the models to single-cell recording data, fMRI data, and human behavioral data.

The method we describe is algorithmic in the sense that it consists of a discrete set of problems that must be solved. In this chapter, we define five specific problems, and we present detailed methods for solving all but one of these problems. These methods are all straightforward and easy to implement. The result is that, after working through this chapter, an interested reader who has an hypothesis about which brain regions may be mediating some behavior should be able to construct a computational version of that model and test its predictions against single-cell recordings, fMRI data, and behavioral data.

Of the five well-defined steps to be solved, the first and most difficult problem is to identify the brain areas that mediate performance in the behavior to be modeled. A necessary part of this process is to specify the interconnections among these areas and whether each projection is excitatory or inhibitory. The second problem is to write a set of differential equations that describe the neural activation in each of these brain regions. A related problem is to solve these (simultaneous nonlinear) differential equations. Solving these two problems results in a computational model that predicts dynamic changes in neural activation in each brain region specified by the model. To model real data, however, some interface needs to be added that describes how the dependent variable of interest is related to the neural activations that are the core currency of the model. The final three problems are directed at constructing these interfaces. Problem 3 is to find an interface to fit single-cell recording data—here we must convert the continuous neural activations obtained from solving the differential equations of Problem 2 into spike trains. The goal of Problem 4 is to fit fMRI data, which requires adding an interface that models the transformation from neural activation to the fMRI Blood Oxygen-Level Dependent (BOLD) signal. Finally, to fit behavioral data, Problem 5 seeks to add assumptions that relate neural activation in a particular brain region to behavior and then derive predictions for the relevant dependent variable (e.g., accuracy or re-
response time). This chapter is organized around these five problems, with a major section devoted to each. After this we close with some general comments.

2.1 PROBLEM 1: IDENTIFYING RELEVANT BRAIN STRUCTURES

The first step in designing a computational neural network is to identify the brain structures that are hypothesized to mediate the performance in the behavior to be modeled. This task includes specifying the interconnections among these areas and whether each projection is excitatory or inhibitory. In most cases, this network is incomplete, in the sense that there generally are certain perceptual or cognitive processes omitted by the network that nonetheless are necessary for the expression of the behavior of interest. For example, a working memory model may specify the neural structures that mediate the maintenance of the memory, but say nothing about the neural networks that mediate the perception of the stimulus or the selection and execution of a motor response.

In general, there are two ways to deal with such omissions depending on whether the omitted processes are upstream from (i.e., precede) or downstream from (i.e., follow) the specified network. In the working memory example, perceptual processes should be upstream from the memory maintenance network and response execution processes downstream. Upstream processes must be directly modeled, in at least a rudimentary fashion, because they provide input to the specified network. Our approach has been to grossly oversimplify such upstream processes and simply model their input to the specified network as either on or off. Several examples of this technique appear throughout this chapter. Downstream processes can generally be ignored except when fitting the model to behavioral data that depend on these processes. In such cases, our approach has been to add a simple cognitive model of the downstream processes as an interface between the specified network and the observed behavior. Examples of such behavioral interfaces are given in the penultimate section of this chapter.

The problem of identifying even a partial neural network is, by
far, the most difficult of the five problems that we discuss in this chapter. It is also the only problem of the five that is not computational; partly because of this qualitative difference, it is the only problem for which we do not provide an algorithm for solving. Because the task of identifying relevant brain areas is difficult and without set rules to follow, our only recourse in the limited space we have here is to provide some general guidelines and suggestions. These will surely prove insufficient, so a realistic goal of this chapter is to provide algorithmic methods for building a computational model of an existing neural network and for fitting that model to single-cell recording, fMRI, and behavioral data.

Perhaps the most important advice we can give is not to expect any simple solution to this problem. For example, it is highly unlikely that a plausible neural network will be identified by studying only a single type of data. Instead, solving this problem requires a thorough review and reconciliation of disconnected areas of the literature in the brain sciences. When proposing a set of neural structures and their interconnections, it is vital to consider evidence from converging methods that include many different levels of analysis, such as neuroimaging, neuropsychology, behavioral neuroscience, neurophysiology, neuroanatomy, and neuropharmacology.

In trying to decide which structures are relevant for the model, a useful approach is to establish which specific lesions impair the behavior of interest. A double dissociation is strong evidence that two behaviors are mediated by two functionally separate brain systems (Ashby & Ell, 2002). A wealth of such data comes from neuropsychological studies of human patients with specific brain lesions caused by various kinds of injury or neurodegenerative disease, such as Huntington’s, Parkinson’s, or Alzheimer’s disease. Temporary disabling of a brain region that is required for a specific function can also be achieved with experimental techniques such as transcranial magnetic stimulation, Wada testing (disrupting activity in one hemisphere), and electrical stimulation or cortical cooling during neurosurgery. However, it is always important to confirm any double dissociation observed in humans with similar dissociations from well-controlled animal lesion studies (Ashby & Ell, 2002).

The techniques that investigate the loss of a given function due to the disruption of a brain region are complemented with techniques
that measure brain activity associated with this function. These methods include electroencephalography (EEG), ERPs, magnetoencephalography, fMRI, positron emission tomography, near-infrared spectroscopy, and single-cell recording. Due to its invasive nature, this latter technique is mostly used in animals. Pharmacological studies can also provide insights into both disruption and enhancement of specific neurotransmitter systems involved in certain behaviors. These kinds of experiments can provide powerful evidence, especially the animal studies that allow for more rigorous and flexible designs. For example, agonistic (or antagonistic) drugs that facilitate (or impair) a behavioral response can be injected into defined brain areas at various specific times within a behavioral paradigm.

The animal literature is highly relevant to understanding the neural basis of human behavior. The goal is to integrate the animal literature, human lesion findings, and results from functional neuroimaging studies. This goal often requires establishing the homology between the animal and human brain. The behavioral tasks given to animals and humans may also differ, even when both investigate the same psychological construct. Therefore, it is possible that some discrepancy in the brain-behavior relation may result from some behavioral or methodological differences. Still, to get the most comprehensive picture, both animal and human literatures should be consulted.

Interconnections among brain areas should be specified according to whether each of the projections is excitatory, inhibitory, or modulatory. The spontaneous basal firing rates can differ between structures as well and are usually associated with their inputs (e.g., structures with high basal firing rates often receive inhibitory inputs). A sophisticated dynamic model may also take into account the length and conducting speeds of axons and/or the number of synapses that occur between two functionally connected areas. The anatomical organization of projections is mapped out in studies using anterograde and retrograde tracing techniques. These findings can be complemented with those from a relatively new technique called functional connectivity analysis (using fMRI), which is useful in identifying functionally related brain areas and distributed networks by means of their synchronous fluctuations in signal intensity.

Completing this laborious chore of literature search to build the most plausible neural network is not only advantageous for increasing
the chances of realistic and valid predictions that may result from the model, but it also provides a great service to the brain sciences community. Separate fields do not communicate enough to gain a more comprehensive picture of the current state of knowledge, although we all know that each technique is weak in isolation, but convergent findings from multiple techniques can be convincing. Therefore, using the most appropriate brain areas and projections to build the architecture of the model is an important step for distilling and clarifying a vast, disconnected literature. In addition, if a neural network is biologically accurate, it can confirm the neuroscience studies when it is tested with specific lesions, and it can also provide an artificial environment for testing various lesions that are impossible or too costly to attain in humans and other animals.

2.2 PROBLEM 2: WRITING THE DIFFERENTIAL EQUATIONS

After the relevant brain areas have been identified and their interconnections specified, the next step is to write a set of equations that describe the neural activation in these regions. The main challenge here is to select an appropriate level at which to direct the equations. If too global a level is chosen, the resulting model will lack biological plausibility, and, as a result, it will be unable to account for neuroscience data. If the model is too detailed, it may account spectacularly for single-cell phenomena, but it is likely to be too complex to account for human behavioral data.

Historically, two separate traditions have constructed computational models of neurons and neural networks. Within psychology, connectionist models have a long and successful history, and they have been used to account for a wide variety of behaviors (Rumelhart & McClelland, 1986). However, the units in connectionist models bear only superficial similarity to real neurons, and there is almost never an attempt to associate units with specific brain areas. Thus, although connectionist models provide powerful descriptions of human behavior, they are not able to account for single-cell recording data or other types of neuroscience data.

A separate discipline called computational neuroscience builds bi-
ologically realistic models that attempt to model biophysical properties of single neurons (e.g., Amini, Clark & Canavier, 1999; Brunel & Wang, 2001; Canavier & Wilson, 1999; Durstewitz, Kelc & Gunturkun, 1999; Durstewitz, Seams & Sejnowski, 2000). Such models, which are often multicompartmental and include Hodgkin-Huxley-like dynamics, are applied to single-cell data, and typically no efforts are made to model behavior. Computational neuroscience models are often highly complex, and in some cases as many as hundreds of differential equations are used to model a single cell. Clearly, even with modern computing methods, this complexity is too high if one’s goal is to model human behavior.

Thus, there is a pressing need for models that lie between the two extremes of the complex yet biophysically realistic models and the connectionist models that are computationally simple but lack neurobiological realism. For this reason, to accomplish our goals, we need to develop a new method of building neural network models that imbues the resulting models with more biological plausibility than is typically found in connectionist networks, but that also results in simpler models than are typically found in computational neuroscience.

The first step is to identify the key functional properties of the neural network that was hypothesized to mediate the behavior of interest (i.e., the network identified in the solution of Problem 1). For example, consider the network illustrated in Figure 2.1. Here the hypothesis is that whatever behavior we are modeling is mediated by a simple circuit consisting of two neural regions. Region A is assumed to receive an excitatory input from some lower-level sensory area and to send an excitatory projection to region B, whereas region B receives an excitatory projection from A and sends an inhibitory projection back to A. The model assumes that the behavior is mediated by these functional interconnections, so these interconnections must be captured in the mathematical equations that model the network. Any other features included in the equations must necessarily increase their complexity and thereby make the task of modeling behavior more difficult.

However, the simplest model of the interconnections shown in Figure 2.1 would have little biological plausibility, and hence could not be used to model neuroscience data. So some extra complexity is nec-
Fig. 2.1: A simple neural network model consisting of two neural regions and an input region. Also shown are the neural activations predicted by the computational model that describes processing in this network.

necessary. The key is to add the minimum amount of complexity needed to model the most important biophysical properties of neurons. Our approach has been to model two key biophysical properties that are shared by all neurons: (a) saturation—every neuron has a maximum firing rate; and (b) decay—if all inputs to a neuron cease, then the activation in that cell will decay to some baseline firing level. In our applications, these biophysical properties, together with the hypothesized interconnections, have been sufficient for modeling single-cell recording data, neuroimaging data, and human behavioral data. Yet as the modeling efforts focus more heavily on neuroscience data, it may be necessary to increase the complexity of the models by trying to model other, more subtle biophysical properties.

We illustrate the approach by writing differential equations that describe the neural activation in regions A and B in the Figure 2.1 network. All of the equations have identical structure, and there is a straightforward algorithm for constructing these equations. In fact, the algorithm is so simple that no special mathematical expertise is required. We begin with region A. The left side of each equation is the derivative of the activation in the region we are modeling. For
region A, we can denote the activation at time $t$ as $A(t)$, in which case the left side becomes

$$\frac{dA(t)}{dt} =$$ \hspace{1cm} (2.1)

An intuitive description of this derivative is that it equals the firing rate of the cells in neural region A. Any excitatory input to region A will increase this firing rate and hence will appear on the right side as a positive term, and any inhibitory input to region A will decrease the firing rate and therefore appear as a negative term on the right side.

We begin with the excitatory input $I(t)$ from the input region. We model this as a simple square wave that has the value $I(t) = 1$ if the stimulus is present and $I(t) = 0$ if the stimulus is absent. More detailed models of the input region could be used, but of course this would complicate the analysis. For example, if the task is primarily cognitive and regions A and B are somewhere in the frontal cortex, it is quite plausible that the neural dynamics in the visual cortex may not have significant effects either on the observable behavior or the neural dynamics in regions A and B. Because the projection from this input area is assumed to be excitatory, $I(t)$ appears on the right side as a positive term.

$$\frac{dA(t)}{dt} = \alpha_A I(t)$$ \hspace{1cm} (2.2)

The constant $\alpha_A$ is a measure of the strength of the synapse between the input area and region A. In most applications, $\alpha_A$ is unknown and must be determined from some data source. Our approach has been to estimate constants such as $\alpha_A$ from single-cell recording data and then leave these constants fixed in all other applications.

The problem with the model so far is that a continual input will cause activation in region A to increase to infinity. As mentioned earlier, one of our goals was to build saturation into our models. This is accomplished in a straightforward fashion by adding a multiplier to the input term

$$\frac{dA(t)}{dt} = \alpha_A I(t)[1 - A(t)]$$ \hspace{1cm} (2.3)

As the activation $A(t)$ increases from 0, the multiplier $1 - A(t)$ decreases, which reduces the excitatory effects of the input.
approaches 1, $1 - A(t)$ approaches 0 and the effect of the constant input is further reduced. Eventually the activation $A(t)$ effectively equals 1. At this point, $1 - A(t)$ equals 0 and the input no longer has any effect. Thus, the $1 - A(t)$ multiplier causes activation to saturate at 1. Adding a multiplier of this type (i.e., 1 minus activation) is a standard technique we use for every excitatory term.

The next step is to account for the inhibitory input from region B. Because this input decreases activation in region A, it carries a negative sign on the right-hand side.

$$\frac{dA(t)}{dt} = \alpha A I(t)[1 - A(t)] - \beta A B(t)$$  \hspace{1cm} (2.4)

As before, the constant $\beta_A$ measures the strength of a synapse, this time between regions B and A. The problem with this method of modeling inhibition is that sustained activation in region B could drive the activation in region A to negative values. To correct this problem, we again add a multiplier:

$$\frac{dA(t)}{dt} = \alpha A I(t)[1 - A(t)] - \beta A B(t) A(t)$$  \hspace{1cm} (2.5)

Now as the inhibitory input $B(t)$ drives the activation $A(t)$ lower and lower, the multiplier $A(t)$ continually decreases, thereby reducing the inhibitory effects. When $A(t)$ is effectively zero, the multiplier is also zero, and continued inhibition has no further effect. Thus, the multiplier $A(t)$ induces a floor at $A(t) = 0$. Adding a multiplier of this type (i.e., equal to the total activation) is a standard technique we use with all inhibitory inputs.

We have now accounted for all inputs to region A. Even so, note that if all inputs to region A are turned off, the change in activation (i.e., the derivative) is zero, meaning that the activation will neither increase nor decrease until an input reappears. Of course, in real neurons, activation would decay under these conditions back to a baseline firing level. If the baseline firing level is zero, then a simple way to model this decay is by adding one more term.

$$\frac{dA(t)}{dt} = \alpha A I(t)[1 - A(t)] - \beta A B(t) A(t) - \gamma A A(t)$$  \hspace{1cm} (2.6)

The constant $\gamma_A$ measures the speed of decay. If the spontaneous firing rate is high, we can define a parameter $A_{base}$ as the baseline
activation level and replace the last term with

\[-\gamma_A [A(t) - A_{\text{base}}].\] (2.7)

Virtually all equations in our models follow this same pattern. The left-hand side is the derivative of the activation. The right side includes a term for each input plus a decay term. Excitatory inputs are preceded by a plus and followed by the multiplier “1 - Activation,” whereas inhibitory inputs are preceded by a minus and followed by the multiplier “Activation.” This method produces differential equations that are nonlinear and whose solutions are bounded between 0 and 1 (or baseline and 1).

Following this method, it is straightforward to write an equation that describes the activation in region B with excitatory input from region A and baseline at 0:

\[
\frac{dB(t)}{dt} = \alpha_B A(t)[1 - B(t)] - \beta_B B(t).\] (2.8)

The simple network shown in Figure 2.1 is therefore modeled by a set of two nonlinear differential equations. Of course, real applications will typically involve more than two brain regions and, therefore, more than two differential equations. Because of their nonlinear and interconnected nature, numerical methods are required to solve these equations in most applications. Either standard algorithms could be programmed in one’s language of choice (e.g., Press, Flannery, Teukolsky & Vettering, 1988) or a high-level language (e.g., MATLAB) with preprogrammed differential equation solvers could be used. For example, Figure 2.1 shows the solutions of Equations 2.6 and 2.8 produced by a standard MATLAB differential equation solver.

Each equation that results from following this algorithm has a free parameter for each term on its right-hand side. In most cases, these parameters each have a well-defined biological interpretation—either as a measure of the strength of a synapse between two brain regions or as a rate at which activation decays in the absence of input. Because of their biological foundation, a natural approach is to estimate these parameters from neuroscience data (e.g., single-cell recordings), but from a statistical perspective they can be estimated during any model-fitting exercise. The synaptic strength parameters...
also provide a natural vehicle for modeling learning, because it is widely assumed that many forms of learning are mediated by changes in synaptic strengths (e.g., Grimwood, Martin & Morris, 2001; Martin & Morris, 2002).

2.3 PROBLEM 3: FITTING SINGLE-CELL RECORDING DATA

The model derived in the last section predicts neural activations in specific brain regions. The data that are perhaps closest to these predicted values are single-unit recordings. For this reason, single-cell recording data provide an excellent opportunity to assess the validity of the network proposed in the solution of problem 1 and estimate the numerical constants of the equations that instantiate the model (e.g., the synaptic strengths). For example, in many of our applications, we use single-cell recording data to fix all numerical constants in the differential equations that describe our model. Then in other applications (e.g., to neuroimaging or behavioral data), we leave all these constants fixed at these same values. Thus, the only free parameters in these later applications are in the unique interfaces we use to convert the neural activations predicted by the model to these new dependent measures (e.g., the fMRI BOLD signal, or percent correct in a behavioral study).

When fitting the models to single-cell recording data, however, one must be careful not to over-fit the models. This is because there is no reason to expect that any specific spike train appearing in a published article represents a gold standard. Almost invariably, moving the electrode a tiny amount uncovers a neuron with a different firing profile, sometimes profoundly different. For this reason, we rarely try to minimize a quantitative fit measure when fitting our models to single-cell recording data. Instead, our goal is typically only to roughly adjust the numerical constants so that the model captures the major qualitative properties of the data.

Although the differential equations describing the model are closely related to single-cell recording data, they are not identical. This is because, as we saw in the preceding section, the solutions of Eqs. 2.6 and 2.8 are continuous functions (i.e., see Figure 2.1), whereas single-
cell recordings produce spike trains. Thus, we need an interface (or model) that converts continuous changes in activation into spike trains. This section describes three different methods, of increasing complexity, for solving this problem.

2.3.1 Poisson Process Approach

In many experiments that we may wish to model, the sensory conditions do not change quickly. For example, in delayed response tasks, a stimulus is shown for a second or two and then a prolonged delay period occurs during which no stimuli are presented. Figure 2.1 shows that when a stimulus of constant intensity is presented, activation in units that receive sensory input that is driven by this presentation quickly ramps up to a steady-state value, where it remains until the stimulus is withdrawn. At this point, activation decays to a new steady-state value. For many behaviors of interest, performance is mostly mediated by these steady-state values. The Poisson process approach to generating spikes is to solve for these steady states and then use these values to generate interarrival times by sampling from a Poisson process.

The first step is to solve for the steady states, which are just the equilibrium-level solutions of the differential equations that describe the model. For example, consider a simple cell that has a baseline activation level of $A_{\text{base}}$ and receives a single input $I(t)$. Following the methods described in the preceding section produces the differential equation

$$\frac{dA(t)}{dt} = \alpha I(t) - \beta [A(t) - A_{\text{base}}]. \quad (2.9)$$

By definition, equilibrium or steady-state behavior occurs when the activation is no longer changing—that is, when $A(t)$ equals a constant value (call this $A$) and the derivative is zero. So to find the equilibrium level solution of this equation, we simply set the derivative to zero and solve for the unknown constant value $A$. In other words, we must solve the following equation for $A$:

$$0 = \alpha I(t) - \beta (A - A_{\text{base}}). \quad (2.10)$$

Of course, this equation has a solution only if $I(t)$ is constant. There are two obvious possibilities—one in which the input has a constant
nonzero value \( I(t) = I \) and one in which no input is presented \( [I(t) = 0] \). In the former case, the solution is

\[
A = \frac{\alpha}{\beta} I + A_{\text{base}}, \tag{2.11}
\]

whereas in the latter case, the solution is \( A = A_{\text{base}} \). When the activation equals these values under the respective input conditions, it will no longer increase or decrease until the input changes. Note that although this is among the simplest possible models (i.e., one cell receiving input), it still makes some rather nonobvious predictions—namely, that the steady-state firing rate of the cell depends not only on the intensity of the input and the strength of the synapse with the input unit, but also on the baseline firing level and the cell’s own decay rate (i.e., \( \beta \)).

The equilibrium-level solutions describe the steady-state activation levels in the cell. One could expect that the firing rate of the cell should be closely related to these levels. The Poisson process approach assumes that spike trains that are representative of this model can be generated from Poisson processes in which the Poisson rate is proportional to the steady-state activation level. A Poisson process is a stochastic process in which the times between successive events are independent samples from identical exponential distributions (e.g., Cox & Miller, 1965).

Figure 2.2 illustrates how a Poisson process can be used to generate spike trains for the model described in this section. To begin, samples \( X_i \) are drawn from a uniform \((0, 1)\) distribution. These are then converted to samples \( T_i \) from an exponential distribution with rate \( \lambda \), where \( \lambda \) is proportional to the steady-state activation level via the transformation (e.g., Ashby, 1992)

\[
T_i = -\lambda^{-1}\log_e(1 - X_i). \tag{2.12}
\]

A series of spikes is then generated with the \( T_i \) defining the interspike intervals.

The Poisson process approach has the advantage of simplicity because it replaces differential equations with linear equations. However, it has the major disadvantage of ignoring the temporal dynamics that are a fundamental property of the differential equation models.
Fig. 2.2: Schematic illustrating a method for simulating a Poisson process.

In many cases, these dynamics are critical for understanding the behavior that the network is mediating. In such cases, the Poisson process approach is unacceptable. Instead, some dynamic approach for generating spikes is needed. A number of such approaches is possible.

2.3.2 Integrate-and-Fire Approach

The field of computational neuroscience has developed standard dynamic methods for generating spikes from continuous-activation models of single cells. The most common approach is to incorporate a so-called integrate-and-fire model (e.g., Koch, 1999). The idea is to integrate the activation and continually compare this integral to a
Fig. 2.3: Schematic illustrating the integrate-and-fire method for producing spike trains.

threshold. When the threshold is exceeded, a spike is generated and the integral is reset to zero and the process repeats. This process is illustrated in Figure 2.3. White noise is typically added during the integration process. Thus, this model adds two free parameters to the model of the single cell—a threshold for generating spikes and the variance of the noise process.

The integrate-and-fire model, without the thresholding process, is described by the equation

\[
\frac{dA_{\text{out}}(t)}{dt} = \alpha A_{\text{in}}(t) + \varepsilon(t),
\]  

(2.13)

where \(\varepsilon(t)\) is a white noise process and \(A_{\text{in}}(t)\) is the neural activation predicted by the single-cell model (i.e., the solution of the differential equation derived in the solution of problem 2). This is the equation of a nonstationary Wiener process. The nonstationary component is provided by the term \(\alpha A_{\text{in}}(t)\).

The integrate-and-fire model is an incomplete model of neural activity because it assumes that the input is perfectly integrated. In real cells, input activation gradually leaks out through the cell’s porous membrane. A popular generalization of the integrate-and-fire model, which corrects this shortcoming, is the leaky integrate-and-fire model (e.g., Koch, 1999). In this model, the process for generating spikes is the same, except the integrate-and-fire equation is replaced with

\[
\frac{dA_{\text{out}}(t)}{dt} = \alpha A_{\text{in}}(t) + \varepsilon(t) - \beta A_{\text{out}}(t).
\]  

(2.14)
The last term makes the model leaky because if the input stops, output activation gradually decays (or leaks) to zero.

The leaky integrate-and-fire model is closely related to the classic Ornstein-Uhlenbeck stochastic process (Cox & Miller, 1965). In fact, if \( A_{in}(t) = 0 \), they are equivalent. So the integrate-and-fire model could be seen as adding a Wiener process to the intracellular activation, whereas the leaky integrate-and-fire model adds an Ornstein-Uhlenbeck process.

If the methods described in the solution of problem 2 are followed when constructing the differential equation models of each neural unit, then there is no need to incorporate the decay term from the leaky integrate-and-fire model. This is because an identical decay term is already built into each equation. Thus, the models derived previously are inherently leaky. As a result, the simpler integrate-and-fire model can be used to generate spikes.

### 2.3.3 Dynamic Integrate-and-Fire Approach

The integrate-and-fire models work well in many applications, but they suffer from one flaw that becomes fatal in any application where the timing of spikes is critical. For example, consider the model shown in Figure 2.4. Note that there are two separate projections from region A to region C—a direct pathway and an indirect pathway through region B. In real cells, of course, the indirect path might take longer to traverse than the direct path, but the differential equations that would be used to model this network predict that activation rises immediately everywhere in all pathways at the same time. Therefore, the integrate-and-fire models also predict that the indirect path will generate spikes in region C at (almost) the same time as the direct path. Thus, any behavior that depends on a temporal difference between these two paths is described inadequately by the integrate-and-fire models.

To correct this problem, we have developed a dynamic version of the integrate-and-fire model. The method is illustrated for the network shown in Figure 2.1, except now we assume all projections are excitatory. This revised network is shown in Figure 2.5. Consider an experiment in which the input is set to some constant value for a short duration and then turned off. The first step is to set the output
Fig. 2.4: A neural network with three regions that has two separate pathways from region A to region C.

activations from regions A and B to zero. Thus, initially, region A receives a nonzero input only from the input region. Next, activation in region A is integrated until a threshold is reached, exactly as in the standard integrate-and-fire model. When the threshold is exceeded, a square wave input is sent from region A to region B. Region B now has a nonzero input, so the activation in B is integrated until a threshold is reached, at which point a square wave input is sent from region B back to region A. This algorithm more closely mimics the natural spike generation process. To make it even more natural, delays can be incorporated to model the time it takes a spike to propagate down the cell’s axon. For example, after activation in region A reaches threshold, a square wave of activation could be delivered to region B after a delay of length \( \Delta_A \), rather than immediately, and a similar delay of length \( \Delta_B \) could be associated with region B. The delays \( \Delta_I \) (I = A or B) would increase with the length of the axons of the cells in region I and decrease with the amount of myelinization. A simple way to add noise to this model is to make \( \Delta_A \) and \( \Delta_B \) random variables. A particularly attractive choice would be to determine \( \Delta_A \) and \( \Delta_B \) by drawing samples from an exponential distribution.

Figure 2.5 also shows an example of this spike-generation method with \( \Delta_A \) and \( \Delta_B \) set to zero. Note the spike-like nature of the activation produced in region B. Region B produces more spikes than region A because it is further removed from the artificial square-wave input that we assumed for the input region. Also note that, despite
assuming that $\Delta_A = \Delta_B = 0$, the method produces temporal delays before the first spike appears that increase with the number of synapses from the initial input. For example, note that the input activation rises to a constant value at time zero and then there is a short delay before activation rises above zero in region A, and a longer delay before the same thing occurs in region B. More realistic spike trains can be produced by recording the times when the threshold is exceeded in each cell and then simply plotting a spike at each of these times.

\section*{2.4 Problem 4: Fitting Neuroimaging Data}

Within the field of cognitive neuroscience, an extremely important challenge for future computational models is to provide quantitative fits to neuroimaging data. In humans, the most common indirect measures of neural activation are made using fMRI (e.g., Bandettini et al., 1992; Kwong et al., 1992; Ogawa et al., 1992). We say \textit{indirect} because fMRI measures the so-called BOLD signal, rather than neural activation (Ogawa, Lee, Kay & Tank, 1990a; Ogawa, Lee, Nayak & Glynn, 1990b). Although it is commonly assumed that the BOLD...
signal increases with neural activation, it is known that the BOLD response is much more sluggish than the neural activation that is presumed to drive it. As a result, for example, the peak of the BOLD signal lags considerably behind the peak neural activation (i.e., by about 6 s).

In the computational models described here, the solutions of the differential equations yield continuous neural activations. Logothetis and colleagues have argued that the BOLD signals most likely are driven by the local field potentials, rather than by the spiking output of individual cells (Logothetis, 2003; Logothetis, Pauls, Augath, Trinh & Oeltermann, 2001). Local field potentials integrate the field potentials produced by small populations of cells over a sub-millimeter range, and they vary continuously over time. Thus, they are most closely related to the direct solutions of the differential equations that define our models (rather than, say, to the spike trains derived in the preceding section).

2.4.1 Linear Models of the BOLD Response

Because the fMRI BOLD signal is related to oxygen levels in the blood, a model of the transformation from neural activation to the fMRI BOLD signal is necessary to fit fMRI data. Several such models have been proposed. Almost all current applications assume that this transformation can be modeled as a time-invariant linear system. Although it is becoming increasingly clear that the transformation is, in fact, nonlinear (e.g., Boynton, Engel, Glover & Heeger, 1996; Buxton & Frank, 1998; Vazquez & Noll, 1998), it also appears that, under the appropriate conditions, these departures from linearity are not severe. For example, the approximation to linearity is apparently improved if the intertrial interval exceeds one second and if brief, tachistoscopic exposure durations are avoided (Vazquez & Noll, 1998). These two conditions are typically met in fMRI studies of high-level cognition.

The behavior of any linear system is completely characterized by its so-called impulse response function. For example, given this function, predicted BOLD signals could be computed for a given brain region by numerically convolving the neural activation that a computational model predicts in that region with the impulse response function (e.g., Chen, 1970). In the fMRI literature, the impulse response
function that characterizes the transformation from neural activation to BOLD signal is known as the hemodynamic response function (hrf). Many models of the hrf have been proposed (Boynton, Engel, Glover & Heeger, 1996; Clark, Maisog & Haxby, 1998; Cohen, 1997; Dale & Buckner, 1997; Friston, Holmes & Ashburner, 1999; Friston, Josephs, Ress & Turner, 1998; Zarahn, Aquirre & D’Esposito, 1997).

One particularly attractive choice is to model the hrf at time $t$, $h(t)$, as a weighted linear combination of basis functions (Friston, Josephs, Ress & Turner, 1998):

$$h(t) = \alpha_1 b_1(t) + \alpha_2 b_2(t) + \alpha_3 b_3(t),$$  \hspace{1cm} (2.15)

where $b_i(t)$ is the $i$th basis function and $\alpha_i$ is its weight. Friston et al. (1998) suggested gamma probability density functions for the basis functions with means and variances equal to 4, 8, and 16, respectively. Thus,

$$b_1(t) = \frac{1}{3!} t^3 e^{-t},$$  \hspace{1cm} (2.16)

$$b_2(t) = \frac{1}{7!} t^7 e^{-t},$$  \hspace{1cm} (2.17)

and

$$b_3(t) = \frac{1}{15!} t^{15} e^{-t}.$$  \hspace{1cm} (2.18)

These three functions were selected to model peaks during the early, intermediate, and late components of the anticipated BOLD signal. The weighting parameters $\alpha_1$, $\alpha_2$, and $\alpha_3$ might vary across participants and/or brain regions.

Once an hrf is selected, the predicted BOLD signal in a specified region of interest is computed by convolving the predicted neural activation in that region, $A(t)$ (obtained from solving the differential equation describing activation in that region) with the hrf:

$$\text{BOLD}(t) = \int_0^t h(\tau) A(t - \tau) d\tau.$$  \hspace{1cm} (2.19)

In our applications of Equation 2.19, we have first fixed the values of all unknown constants in the differential equations by fitting the model to single-cell recording data (using the methods of the previous section). Thus, in our applications, the only free parameters
in Equation 2.19 are in the description of the hrf (e.g., the $\alpha_i$ in Equation 2.15). These parameters can be estimated using standard optimization algorithms, but the use of the basis function hrf model of Equation 2.15 greatly simplifies this process. For example, note that substituting Equation 2.15 into Equation 2.19 produces

$$\text{BOLD}(t) = \alpha_1 \int_0^t b_1(\tau) A(t-\tau) d\tau + \alpha_2 \int_0^t b_2(\tau) A(t-\tau) d\tau + \alpha_3 \int_0^t b_3(\tau) A(t-\tau) d\tau.$$  

(2.20)

If the parameters of the differential equations are fixed using other data (e.g., single-cell recording data), then each of these integrals involves no free parameters. As a result, if we define

$$x_i(t) = \int_0^t b_i(\tau) A(t-\tau) d\tau,$$  

(2.21)

then

$$\text{BOLD}(t) = \alpha_1 x_1(t) + \alpha_2 x_2(t) + \alpha_3 x_3(t).$$  

(2.22)

Thus, the three convolutions specified by Equation 2.21 can be computed numerically, which determines the three vectors $x_i(t)$. The parameters $\alpha_i$ can now be determined using standard linear regression techniques.

Despite the absence of computational models that make specific predictions about neural activation, there has nevertheless been a pressing need to test specific a priori predictions about the BOLD signal. In the absence of a detailed model, the only predictions that can typically be made are that a brain region involved in mediating some cognitive task should be active throughout the duration of that task. This idea can be modeled with a simple neural activation function that equals a constant value during the critical experimental period that defines the task (typically the time between stimulus
presentation and response) and zero otherwise (e.g., Friston et al., 1995). More formally, this function $R(t) = 1/T$ for $0 < t \leq T$, and $R(t) = 0$ otherwise, where $T$ is the assumed duration of neural activation. Because of its shape, $R(t)$ is called the boxcar function. An example is shown in Figure 2.6. To generate predictions from the boxcar model, one simply convolves the boxcar function with an hrf (i.e., equivalent to substituting $R(t)$ for $A(t)$ in Equation 2.19). An obvious and important first test of any computational model derived using the methods described in this chapter is to compare its ability to account for fMRI data with the traditional boxcar model.

As an example of this process, we briefly describe an application by Ashby, Ell, Valentin, and Casale (2003, 2005). Using the procedures described in this chapter, Ashby et al. (2005) derived a neurobiologically plausible model of working memory called FROST (FROntal Striatal Thalamic loops). The parameters in the differential equations that described the model were estimated by roughly fitting FROST to published single-cell recording data (Constantinidis & Steinmetz, 1996; Fuster, 1973; Fuster & Alexander, 1973; Hikosaka, Sakamoto & Sadanari, 1989; Mushiake & Strick, 1995). Using these
fixed parameter values, FROST was then fit to fMRI data from a spatial delayed response task reported by Ell and colleagues (2003). During the stimulus presentation phase of this study, a target stimulus (i.e., a dot) was presented in a certain spatial location for 2.5 s (i.e., the repetition time, or TR, of the scanner). The target dot then disappeared for a delay period of either 2.5 or 5 s. During the response phase, a probe dot appeared in either the same or a different spatial location. The subject’s task was to indicate with a button press whether the location of the probe dot was identical or different from the location of the target dot.

Ell and colleagues (2003) reported a sustained BOLD response during the delay period of this task in several areas predicted by FROST—namely, the posterior parietal cortex (Broadman Area 7), dorsolateral prefrontal cortex (Broadman Areas 9 and 46), the head of the caudate nucleus, the internal segment of the globus pallidus, and the medial dorsal nucleus of the thalamus. Figure 2.7 shows the observed BOLD signal in the five brain regions identified by FROST, along with the predicted BOLD signals generated by convolving the neural activations predicted by FROST with an hrf (the three-parameter delayed gamma function proposed by Boynton et al., 1996). As can be seen, FROST captures the major qualitative properties of the observed data.

As a further test, Ashby and colleagues (2003) compared the fMRI predictions of FROST with those of the traditional boxcar model (shown in Figure 2.6). First, of course, unlike a model like FROST, the boxcar model makes no predictions about which brain regions should be active during the delay period, nor does it make any predictions about single-cell recording data. Thus, models like FROST have a number of important advantages over the boxcar model even before simple goodness-of-fit is considered. In fact, the only asset of the boxcar model is its assumed ability to account quantitatively for BOLD signals in brain regions that have been somehow identified using other means. Even so, this is a significant accomplishment, and it is important to ask whether models developed using the methods described in this chapter can match this ability of the boxcar model.

To answer this question, Ashby and colleagues (2003) compared the ability of FROST and the boxcar model to account for the observed BOLD signals reported in Ell et al. (2003) in the brain regions
identified by FROST as being important in maintaining the working memory representation of the target location. The only free parameters were from the hrf (because the boxcar function has no free parameters, and the parameters of FROST were fixed from the fits to single-cell recordings), and both models used the same hrf.

For both models, the parameters of the hrf were estimated using a least squares criterion. The best-fitting version of the boxcar model accounted for 61% of the variance in the observed BOLD signals, whereas the best-fitting version of FROST accounted for 93% of the variance. Thus, for these data at least, FROST not only made accurate a priori predictions about which brain regions should show significant delay-related activation, but it also gave a better quantitative account of the BOLD signals in these regions than the widely used boxcar model.

2.4.2 Nonlinear Models of the BOLD Response

According to the linear systems approach described by Equation 2.19, the response to a pair of stimuli presented simultaneously should
equal the sum of the responses to each stimulus presented in isolation. This is the well-known superposition property that characterizes all linear systems. As mentioned earlier, superposition is approximately satisfied if the intertrial interval exceeds one second and if brief stimulus exposure durations are avoided (Vazquez & Noll, 1998). However, if stimulus events quickly follow one another, or if brief stimulus exposure durations are used, then it is well documented that the BOLD signal exhibits significant nonlinearities (Hinrichs et al., 2000; Huettel & McCarthy, 2000; Ogawa et al., 2000; Pfeuffer et al., 2003).

One reason these nonlinearities apparently occur is that, although blood flow increases with neural activity, the BOLD signal has a nonlinear dependence on flow (e.g., Mechelli et al., 2001; Miller et al., 2001). For example, the BOLD signal might saturate at high levels of blood flow, in the sense that further increases in flow cause negligible increases in the concentration of deoxyhemoglobin. In this way, a moderately strong stimulus could evoke a near-maximal fMRI response, leaving little room for further increases in response (even to a stronger stimulus).

There have been a number of attempts to model nonlinearities in the BOLD response. One of the earliest and best-known nonlinear models is the balloon model of Buxton et al. (1998), which is based on the biomechanical properties of the brain’s vasculature. The BOLD signal depends on blood flow, blood volume, and blood oxygenation, and the balloon model incorporates the conflicting effects of dynamic changes in both blood oxygenation and blood volume, and it assumes that the volume flow out of the system depends on a balloon-like pressure of the vasculature. For example, when the blood flow is high, the walls of the blood vessels are under greater tension; as a result they push the blood out with greater force, which reduces the rate at which oxygen is extracted from the hemoglobin. The balloon model directly models this inherent nonlinearity. However, the balloon model makes the simplifying assumption that there is no capillary contribution to the BOLD signal—an assumption challenged by more recent models (e.g., Zheng et al., 2002).

One practical drawback to the balloon model is its complexity. Implementing the model requires a major computational effort. Fortunately, computationally simpler alternatives exist. One especially attractive choice is to construct a Volterra series to model the nonlin-
earieties in the BOLD signal. An important theorem in nonlinear systems theory states that the output of almost any time-invariant nonlinear system can be expressed as a Volterra series of its input (e.g., Schetzen, 1980). In the present case, this means that the BOLD response can be defined by the following function (i.e., the Volterra series) of the neural activation \( A(t) \):

\[
\text{BOLD}(t) = \sum_{i=1}^{\infty} H_i[A(t)],
\]

(2.23)

where

\[
H_i[A(t)] = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_i(\tau_1, \cdots, \tau_i) A(t-\tau_1) \cdots A(t-\tau_i) d\tau_1 \cdots d\tau_i.
\]

(2.24)

The function \( h_i(\tau_1, \cdots, \tau_i) \) is called the \( i \)th Volterra kernel.

Note that

\[
H_1[A(t)] = \int_{-\infty}^{\infty} h_1(\tau_1) A(t-\tau_1) d\tau_1,
\]

(2.25)

which is the familiar convolution integral (i.e., see Equation 2.19), so \( h_1(\tau) \) is just the hrf. In other words, \( H_1[A(t)] \) models the linear response and the higher order kernels model the nonlinearities in the response. So the standard linear approach is to assume that \( \text{BOLD}(t) = H_1[A(t)] \). Friston and colleagues (1998) argued that the nonlinearities modeled by the balloon model could be approximated by

\[
\text{BOLD}(t) = H_1[A(t)] + H_2[A(t)].
\]

(2.26)

Thus, we need only supplement the usual linear approach by adding a second-order correction for nonlinearity.

Now

\[
H_2[A(t)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h_2(\tau_1, \tau_2) A(t-\tau_1) A(t-\tau_2) d\tau_1 d\tau_2,
\]

(2.27)

so to follow this approach we must specify the second-order Volterra kernel \( h_2(\tau_1, \tau_2) \). Friston et al. (1998) suggested

\[
h_2(\tau_1, \tau_2) = \sum_{i=1}^{3} \sum_{j=1}^{3} \beta_{ij} b_i(\tau_1) b_j(\tau_2),
\]

(2.28)
where the $b_i(\tau)$ are the gamma basis functions of Eqs. 2.16, 2.17, and 2.18, and the $\beta_{ij}$ are free parameters. The computational advantage of using basis functions to model the hrf extends to the second-order Volterra kernel because substituting Equation 2.28 into 2.27 yields
\[
H_2[A(t)] = \sum_{i=1}^{3} \sum_{j=1}^{3} \beta_{ij} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} b_i(\tau_1)b_j(\tau_2)A(t-\tau_1)A(t-\tau_2)d\tau_1d\tau_2.
\]
(2.29)

If the parameters of the core neural network are estimated using other data, then none of these (nine) double integrals involves any free parameters. Thus, as in the linear case, they can be computed numerically before the parameter estimation process begins. As before, parameter estimation can now be accomplished using standard regression techniques.

Figure 2.8 shows the predicted BOLD response from this model to a simple neural activation (also shown). To generate this prediction, all nonlinear coefficients were set to zero, except $\beta_{11}$, which was set to 4. Also shown is the linear model (i.e., Equation 2.20) that provides the best fit to the nonlinear BOLD response. Although the linear model shows systematic deviations from the nonlinear model, the fit is reasonably good and would probably be satisfactory for most cognitive experiments. With more complex neural activations, the difference between the linear and nonlinear predictions would increase. For example, as mentioned earlier, when stimulus events closely follow one another, the nonlinearities in the BOLD response become especially pronounced.

2.5 PROBLEM 5: FITTING BEHAVIORAL DATA

The models derived in this chapter predict how neural activations change in specific brain regions under different experimental conditions. Of course, neural activations are not behaviors, so to fit the models to behavioral data, some assumptions must be added that describe how neural activation is related to behavior. In most cases, this process involves at least two steps. The first is to identify which brain regions in the hypothesized network control the behavioral response. This problem is similar to problem 1 in the sense that neither has an algorithmic solution. Instead, a solution depends on one’s
knowledge of the relevant neuroscience literatures. For example, the FROST model of working memory (Ashby et al., 2005) includes differential equations that describe neural activation in parietal cortex, prefrontal cortex, thalamus, caudate nucleus, and globus pallidus. Yet it makes sense, for a variety of reasons that are beyond the scope of this chapter, to assume that the integrity of a working memory depends primarily on neural activation within prefrontal cortex.

Once the critical neural structure is identified, the second step is to decide how activation in this region controls behavior. There are many possibilities, but we have had success with two especially simple assumptions. The first assumes that the behavioral response is correct if the activation on a critical unit exceeds a threshold. This is an obvious model for any kind of task that requires a YES/NO or GO/NO GO response. For tasks in which subjects must select among several alternatives, a natural assumption is that different
neural units within the critical region control different responses, and the unit with the greatest activation determines the behavioral response.

2.5.1 Threshold Model

As mentioned before, in tasks requiring a YES/NO or GO/NO GO response, it is natural to assume that a YES or GO response occurs if activation on some critical neural unit exceeds a threshold. Of course, all real data have inherent variability. If accuracy is the dependent variable of interest, then experimental conditions are arranged so that subjects do not always emit the same response to the same stimulus; if the focus is on response time (RT), then the data exhibit trial-by-trial variability no matter what the experimental conditions. The models highlighted in this chapter can only account for such variability if we somehow add noise to the network.

There are many ways to add noise to the models proposed here. Unfortunately, there is a tradeoff among these methods between intuitive plausibility and mathematical tractability. The most intuitively appealing method is to add noise to the input, and perhaps also within each unit, and let it propagate through the network. This makes the differential equations stochastic. Although there is a well-developed mathematical theory of stochastic differential equations (e.g., Øksendal, 2000), the computational problems associated with finding numerical solutions are extreme—mostly because of the discontinuous nature of standard (white) noise processes.

A second choice that is less intuitively appealing, but more mathematically tractable, is to allow the network to be deterministic and then add a stochastic noise process to the output of the critical neural unit that determines the response. For example, suppose we assume that the subject responds YES if the activation $A(t)$ (obtained by solving the differential equation describing the critical unit) exceeds a threshold $\tau$ sometime before a response deadline occurs. If a noise process $\varepsilon(t)$ is added to $A(t)$, then the decision rule becomes

$$\text{Respond YES if } A(t) + \varepsilon(t) > \tau. \quad (2.30)$$

A variety of different models can be used for the noise process $\varepsilon(t)$, including white noise, a Wiener process, or an Ornstein-Uhlenbeck
Computational Cognitive Neuroscience

process (e.g., Cox & Miller, 1965). In any case, however, the resulting stochastic process $A(t) + \varepsilon(t)$ is nonstationary (i.e., nonergodic or inhomogeneous) because $A(t)$ changes with time, which changes the statistical properties of the process. As a result, analytic predictions of either accuracy or RT will be difficult to derive. Even so, some excellent numerical methods for deriving predictions have been developed (e.g., Diederich & Busemeyer, 2003; Smith, 2000), so this method of adding noise is a viable option.

A third method for adding noise, which is the most psychologically implausible of the three, but which leads to simple analytic predictions, is to assume that on each trial a single sample from a noise distribution is added to the output activation of the critical unit. Thus, the decision rule in Equation 2.30 becomes

$$\text{Respond YES if } A(t) + \varepsilon > \tau, \quad (2.31)$$

where $\varepsilon$ is a sample from some noise distribution. A common approach would be to assume that $\varepsilon$ is normally distributed with mean 0 and variance $\sigma_\varepsilon^2$. In this case, analytic predictions about accuracy and RT can be derived in a straightforward manner.

To begin, note that the noise $\varepsilon$ has no effect on the shape of the activation function $A(t)$. This is because $\varepsilon$ has a single value on each trial, so the effect of adding noise is to displace the activation function $A(t)$ up or down on each trial by a random amount. For this reason, the maximum value of $A(t) + \varepsilon$ occurs at the same time point on every trial (i.e., assuming the same stimulus conditions). Call this time $t_R$. Thus, if $A(t) + \varepsilon$ ever exceeds the threshold $\tau$, it must exceed the threshold at time $t_R$. Thus,

$$P(\text{YES}) = P[A(t) + \varepsilon > \tau], \text{ for some time } t$$

$$= P[A(t_R) + \varepsilon > \tau]$$

$$= P[\varepsilon > \tau - A(t_R)]$$

$$= P \left[ Z > \frac{\tau - A(t_R)}{\sigma_\varepsilon} \right]$$

$$= 1 - P \left[ Z \leq \frac{\tau - A(t_R)}{\sigma_\varepsilon} \right], \quad (2.32)$$

where $Z$ has a standard z-distribution. Thus, this probability can be computed from z-tables. If all parameters needed to compute $A(t_R)$
Fig. 2.9: Some classic working memory span data and fits by the FROST model using the threshold model as an interface.

were estimated from previous fits to neuroscience data, the only free parameters here are $\tau$ and $\sigma^2$.  

We (Ashby et al., 2005) previously used this threshold model to fit the FROST model of working memory to the most classic of all working memory span data—namely, the data that formed the basis of Miller’s famous magical number $7 \pm 2$. Figure 2.9 (adapted from Batchelder, 2000) summarizes data from three very different working memory experiments—classic memory span (Guilford & Dallenbach, 1925), span of apprehension or object enumeration (Mandler & Shebo, 1982), and absolute identification of pure tones (Pollack, 1952). Figure 2.9 shows the proportion of correct responses as a function of set size for all three experiments. The magical number seven results if the criterion for working memory capacity is defined as the number of items for which accuracy is 50%.

Without going into details of the application, we (Ashby et al., 2005) assumed that each to-be-remembered item would be encoded by activity in its own prefrontal cortical unit, and that the subject would
correctly retrieve this item if the activation in its associated unit exceeds a threshold $\tau$ at the time of retrieval. The free parameters in this decision model are $\tau$ and the noise variance $\sigma^2$. Fits of this model to the classic data shown in Figure 2.9 are indicated by the solid curve.\(^1\) As can be seen, FROST provides an excellent account of these behavioral data, accounting for 99.7% of the variance in the mean memory span data.

The threshold model can also be used to make RT predictions, although in the case of RT, the stochastic noise model described by Equation 2.30 is usually a better alternative. In some simple cases, however, the static noise model may suffice. The derivation here closely follows that presented by Ashby (1982). We assume that a YES response occurs as soon as the threshold $\tau$ is first exceeded. Thus, the cumulative RT distribution function on YES trials is given by

$$
P(\text{RT} \leq t | \text{YES}) = \frac{P[A(t) + \varepsilon > \tau, A(t_R) + \varepsilon > \tau]}{P[A(t_R) + \varepsilon > \tau]}$$

$$= \frac{P[A(t) + \varepsilon > \tau]}{P[A(t_R) + \varepsilon > \tau]}.$$  \hspace{1cm} (2.33)

Now the denominator of Equation 2.33 is given in Equation 2.32, whereas the numerator is equal to

$$P[A(t) + \varepsilon > \tau] = 1 - P \left[Z \leq \frac{\tau - A(t)}{\sigma} \right].$$  \hspace{1cm} (2.34)

Equations 2.32 to 2.34 specify the RT distribution in any experimental condition in which activation functions can be predicted.

### 2.5.2 Max Activation Model

The threshold model makes sense only if there are two response alternatives. In experiments with more than two alternatives, a plausible assumption is that different units control different responses, and that

\(^1\)A third parameter was estimated in these fits. This extra parameter measured the strength of lateral inhibition between prefrontal cortical units. This parameter could not be estimated from single-cell recording data because the available single-cell data were all derived from tasks in which there was only one item in the memory set.
the behavioral response is determined by the unit with the greatest activation. As with the threshold model, a variety of different noise models could be used. However, the computational problems with the more psychologically plausible models are even more severe in the max activation model because of the greater number of response alternatives. Even so, under appropriate distributional assumptions, the single-noise-sample-per-trial model remains tractable.

Suppose there are \( n \) alternative responses. Let \( A_i(t) \) denote the output activation of the unit that controls response \( i \), let \( \varepsilon_i \) denote the value of the noise added to \( A_i(t) \), and let \( t_R \) denote the response time. Then the max rule is to

\[
\text{Respond } i \text{ if } A_i(t_R) + \varepsilon_i = \max_k [A_k(t_R) + \varepsilon_k],
\]

Computing the probability that response \( i \) is emitted is, in general, a difficult problem. However, the solution is well known in the special case in which the \( \varepsilon_k \) are all independent and identically distributed with double exponential distributions (Yellott, 1977). In this case, the probability reduces to the familiar Luce-Shepard choice rule (Luce, 1963; Shepard, 1957):

\[
P(i) = \frac{A_i(t_R)}{\sum_k A_k(t_R)}.
\]

A slight generalization of this model, which adds one free parameter, is to assume that the probability of response \( i \) is given by

\[
P(i) = \frac{A_i^\gamma(t_R)}{\sum_k A_k^\gamma(t_R)}.
\]

The exponent \( \gamma \), which was introduced to the model by Ashby and Maddox (1993), is inversely related to response variability. For example, as \( \gamma \) approaches infinity, the model predicts that the subject will respond \( i \) on all trials for which the activation is greatest on unit \( i \). In contrast, when \( \gamma = 0 \), the model responds at chance performance no matter what the neural activations. For this reason, \( \gamma \) is inversely related to the amount of internal noise, \( \sigma_\varepsilon^2 \) (i.e., see Ashby & Maddox, 1993).
We (Ashby et al., 2003) have also used the max activation rule to fit the FROST model to some human behavioral data from a task designed to be as similar as possible to the delayed response tasks used in many single-cell recording studies and in the Ell et al. (2003) neuroimaging study. The major constraint was to increase difficulty enough so that errors were common. On each trial, a target dot and 0, 1, or 2 distractor dots of different colors were briefly displayed at random locations. After a delay of 0, 2, 4, or 6 seconds, the target dot and 49 distractor dots of the same color as the target were displayed. The subject’s task was to point to the location of the target dot. Experiment 1 varied the number of initial distractors (i.e., from 0 to 2), whereas Experiment 2 was identical to the one-distractor condition of Experiment 1, except the target had the same color on 75% of the trials.

The proportions of correct responses at each delay are plotted in Figure 2.10 for both experiments. Also shown are the fits of FROST using the max activation model (i.e., Equation 2.37). The critical variable in FROST is the amount of activation in the PFC working memory unit associated with the spatial location of the target dot. The model predicts a correct response so long as this activation is substantially larger than the activations produced by the 49 distractor dots. The data from Experiments 1 and 2 were fit simultaneously (the data had 20 degrees of freedom). As in our other applications, the numerical constants in the basic FROST circuit were determined by first fitting the model to single-cell recording data. As a result, FROST had only four free parameters in this application.\(^2\) As can be seen, FROST provides an excellent account of these behavioral data, accounting for more than 97% of the variance in these 20 data points. This is especially impressive given that only four free parameters were estimated to fit these 20 data points.

\(^2\)The four parameters were as follows: (a) the response variability parameter \(\gamma\) from Equation 2.37, (b) the summed magnitude of PFC activations induced by all 49 distractor dots, (c) the strength of PFC lateral inhibition, and (d) a parameter that modulated the strength of lateral inhibition in Experiment 2 depending on the target selection probability (i.e., so that the inhibition of more probable targets on less probable targets was greater than the opposite inhibition).
Fig. 2.10: (a) Experiment 1 and (b) Experiment 2. Human delayed response data and fits by the FROST model using the max activation model as an interface.
2.6 CONCLUSIONS

The cognitive neuroscience revolution has carried with it a dramatic increase in the variety of data sources that are now routinely considered. Whereas the older purely cognitive theories generally were concerned only with behavioral measures of response accuracy and response time, today’s theories might also be challenged with neuropsychological patient data, functional neuroimaging data, EEGs and ERPs, animal lesion results, and single-cell recording data. Such incredible diversity requires a new approach to model building. The methods described in this chapter were developed as a first attempt to meet this challenge.

This chapter focuses on fitting three different types of data: single-cell recordings, fMRI, and behavioral data. However, the same models could also be used to fit other kinds of data. One important application is to model neuropsychological patient data. For example, lesions can be modeled by setting the output of the lesioned brain region to zero or, perhaps more realistically, by reducing the gain on this output.

If the goal is to model a qualitatively different type of data than was discussed in this chapter, then a new interface is needed that carries the neural activations predicted by the core neural network to this new dependent variable. For example, an obvious extension of the methods presented here would be to ERPs. In traditional ERP analyses, the goal is to solve the so-called inverse problem—that is, to determine which brain regions produced the pattern of electrical charges observed on the scalp. In our case, however, the brain regions and their activations would be known (i.e., specified by the model), so our problem would be to solve the less difficult forward problem— to determine the pattern of electrical charges on the scalp that would follow from a given set of neural activations. The forward problem poses a significant challenge, and a complete discussion is beyond the scope of this chapter. Even so, significant progress has been made toward its solution (e.g., Ferguson & Stroink, 1997), so it seems reasonable to expect that an acceptable interface could be developed that would allow the network models described here to be fit to ERP data.

Another advantage of the models proposed here is that, because
they are closely related to the underlying neuroscience, they are easy to augment with other neuroscience models. For example, we recently proposed a model of how dopamine modulates glutamate activity in frontal cortex (Ashby & Casale, 2003). This model makes much more detailed neuroscience assumptions than the models described in this chapter (e.g., about how dopamine has different modulatory effects on NMDA versus AMPA receptor activations). Models produced using our standard approach could be generalized by incorporating more detailed neuroscience models of this type into the differential equations that describe activation in frontal areas. The resulting augmented model could then be used to predict how changes in cortical dopamine affect the behavior of interest.

In summary, the methods described here represent a new approach to the computational modeling of cognitive processes. Our goal in developing these methods was to bridge the gaps among neuroanatomy, behavioral neuroscience, brain imaging, and cognition. Although it is far too early to decide how successful we have been, it is true that the application of these methods has forced us to consider carefully and attempt to integrate a far more extensive and diverse set of results than in our previous modeling efforts, which used more traditional, purely cognitive methods. In this sense, at least, the approach described here has already been successful.

ACKNOWLEDGMENTS

Preparation of this chapter was supported in part by Public Health Service Grant MH3760. We thank Michael Casale, Shawn Ell, and Elliott Waldron for their invaluable participation in the development of the methods described in this chapter.

References


