In W. Batchelder, H. Colonius, E. Dzhafarov, & J. Myung (Eds.),
New handbook of mathematical psychology, Volume 2 (pp. 223-270).
New York: Cambridge University Press.

Computational Cognitive Neuroscience

F. Gregory Ashby
University of California, Santa Barbara

Contents

6.1 Introduction

Cognitive neuroscience was born in the 1990's amid a technological explosion that produced powerful new methods for noninvasively studying the human brain, including functional magnetic resonance imaging (fMRI) and transcranial magnetic stimulation (TMS). These exciting new technologies revolutionized the scientific study of the mind, giving unprecedented observability into the neural processes that mediate human thought and action. With the new data came a growing need for new kinds of theories that could simultaneously account for the behavioral data that are the bread and butter of traditional mathematical psychology as well as the brain-related measures coming from the new technologies. Computational Cognitive Neuroscience (CCN) was created to fill this void.

CCN evolved from computational neuroscience on one side and connectionism, neural network theory, and machine learning on the other. Like computational neuroscience, CCN strives for neurobiological accuracy and like connec-

tionism, a major goal is to account for behavior. But CCN is unique because most computational neuroscience models make no attempt to account for behavior and most connectionist models make no attempt to be biologically detailed. The biological detail included in CCN models adds many more constraints on the modeling process than more traditional approaches. As a result, two researchers independently modeling the same behavior are more likely to converge on highly similar models with this new approach, and for this reason the resulting models should have a permanence that is unusual with older approaches. Not surprisingly, a growing of number of researchers are now pursuing the endeavor of CCN modeling (e.g., Anderson, Fincham, Qin, & Stocco, 2008; Ashby & Helie, 2011; Frank, 2005; Hartley, Burgess, Lever, Cacucci, & O'Keefe, 2000; O'Reilly, Munakata, Frank, Hazy, et al., 2012).

6.1.1 A Brief History

The field of computational neuroscience became popular with Hodgkin and Huxley's (1952) Nobel Prize winning efforts to model the generation of action potentials in the giant squid axon. Like the Hodgkin and Huxley model, most computational neuroscience models include only a single neuron. A common approach, called compartment modeling, models a neuron's axons and dendrites as cylinders and the soma as a sphere. Next, partial differential equations that describe the propagation of action potentials are written for each of these compartments. A standard application is to model the results of patch-clamp experiments in which current is injected into the neuron at some location and then the intracellular voltage is measured at a variety of locations on the cell. Some compartment models are extremely accurate, but highly complex. For example, some single-neuron models have hundreds or even thousands of compartments (e.g., Bhalla & Bower, 1993; Segev, Fleshman, & Burke, 1989). Historically, computational neuroscience models have almost never tried to account for behavior. In most cases, such a goal is precluded by the complexity of the single-neuron models that are used.

Neural network theory originated in the work of McCulloch and Pitts (1943). However, because the goal quickly became to model or at least simulate behavior, neural network theory diverged from computational neuroscience with the work of Newell, Shaw, and Simon (1958) and Rosenblatt (1958). At that time, computing power was too limited and there simply was not enough known about the neural basis of behavior to support a research program that tried to model behavior in a biologically accurate way. Thus the fields of artificial intelligence and the more modern related field of machine learning place almost all emphasis on behavior and almost none on neuroscience.

Connectionism (Rumelhart & McClelland, 1986) and modern neural network theory (e.g., Haykin, 2009) take an intermediate approach in the sense that biologically plausible properties are often seen as advantages, although they rarely are requirements. Connectionist models have some features in common with the brain – including distributed representation, continuous flow, and the modeling of memory as changes in synaptic strengths. Even so, almost all connectionist models include many features that are now known to be incompatible with brain function. For example, there is generally no attempt to identify units in connectionist models with specific brain regions, and even when there is, there is little attempt to model inputs and outputs to these regions in a biologically accurate way. Similarly, units in connectionist models typically do not behave like real neurons, and the learning algorithms that are used often have little biological plausibility (e.g., backpropagation).

Each of these fields makes important contributions. Computational neuroscience provides a formal framework to test theories of biophysics. Artificial intelligence and machine learning allow engineers to construct machines and algorithms that exhibit intelligent behavior. Connectionism allows psychologists to construct neurally-inspired models of behaviors that are so complex or poorly understood that it would be premature to attempt to build more biologically detailed models. CCN is not meant to supplant these older approaches, but rather to fill a new void that was created by the cognitive neuroscience revolution. There are now many behaviors that are understood well enough at the neural level to permit biologically detailed mathematical modeling. CCN was born in an attempt to exploit these new data.

Because of this motivation, it is not surprising that the field of CCN began shortly after the onset of the cognitive neuroscience revolution during the 1990's. The first break with existing approaches came with attempts to associate nodes in fairly traditional connectionist or neural network models with specific brain regions. This trend toward increased biological detail continued with more biologically plausible learning algorithms, and more realistic models of the individual units (e.g., Ashby, Alfonso-Reese, Turken, & Waldron, 1998; Cohen, Braver, & O'Reilly, 1996; Cohen & Servan-Schreiber, 1992; McClelland, McNaughton, & O'Reilly, 1995). Simultaneously, there were also attempts to formulate general modeling principles of this new approach (Ashby & Helie, 2011; Ashby & Valentin, 2007; O'Reilly, 1998; O'Reilly et al., 2012).

6.1.2 Organization of the Chapter

This chapter is organized as follows. Section 2 describes some advantages of the CCN approach. Section 3 describes some of the CCN principles that guide model development and model testing. Section 4 describes some common approaches used in CCN models in which each unit models a single spiking neuron. Section 5 describes an alternative approach that models the firing rates of large populations

of neurons. Section 6 describes models of learning that are based on the neuroscience literature on long-lasting synaptic plasticity. CCN models make predictions about neural activations in each brain region included in the model. To test the model against empirical data, some modeling interface is typically needed that converts the predicted neural activations to the dependent measure that defines the data. Section 7 describes these interfaces and other relevant issues for applications to single-unit recording data, behavioral data [i.e., accuracy and response time (RT)], fMRI BOLD responses, TMS data, and pharmacological and neuropsychological patient data. Section 8 discusses the problem of parameter estimation and model evaluation, and Section 9 closes with some general comments and conclusions.

6.2 Advantages of CCN Modeling

Most of the advantages of CCN over more traditional purely cognitive mathematical modeling are due to the many neuroscience-derived constraints that CCN models must satisfy – constraints that drastically reduce the space of compatible models. Architecturally, the construction of CCN models is sharply constrained by neuroanatomy and basic neuroscience results. For example, if the model includes a cortical region and the striatum then neuroanatomy tells us that the only possible CCN model of these two regions is one where cortex sends an excitatory projection to the striatum and there is no direct return projection from the striatum to cortex. Restrictions such as these are in sharp contrast with methods used to construct traditional cognitive or connectionist models, where any architecture is allowed, and alternative models are evaluated almost solely on the basis of goodness-of-fit to available RT and accuracy data.

CCN models are constrained not only in architecture, but also in process. For example, neuroanatomy specifies whether each connection is excitatory or inhibitory, and single-unit recording data serve to constrain the dynamics of individual units in the model (e.g., whether the units have a high or low tonic firing rate, whether they fire in bursts or at a steady rate). Furthermore, neuroscience data also sharply constrain how learning and memory are modeled. For example, the evidence is good that dopamine (DA) mediates feedback-driven synaptic plasticity in the striatum (e.g., Doya, 2007; Schultz, 2002). When DA neurons fire, DA is released approximately uniformly throughout large regions of the striatum, and as a result, any CCN model that includes striatal-mediated learning must assume global rather than local learning rules.

The more constrained CCN modeling process confers a number of specific advantages to CCN models. The remainder of this section describes the most important of these.

6.2.1 Testing Against Many Different Data Types

Whereas cognitive models can generally be tested only against RT and accuracy data, CCN models can be tested against many different kinds of dependent measures. Theoretically this should include virtually any dependent measure between behavior at the highest level and single-unit recording data at the lowest level. So for example, the same CCN model could be tested against RTs, accuracies, single-neuron recording data, fMRI BOLD responses, and EEG recordings. In addition, CCN provides a principled method for modeling the effects of various interventions that affect neural processes, including TMS, neuropharmacological treatment, and neurosurgical procedures such as ablation or deep-brain stimulation. Similar approaches could be used to account for behavioral deficits that are associated with various neuropsychological conditions (e.g., Parkinson's disease, Huntington's disease, anterograde amnesia). Requiring successful models to simultaneously account for more different kinds of data necessarily increases model identifiability.

6.2.2 Model Inflexibility

CCN models often include many unknown constants (or parameters) that must be set (or estimated) during the model-fitting process. A traditional cognitive-based mathematical model with the same number of free parameters would be so mathematically flexible that it would be difficult to falsify on the basis of goodness-of-fit alone. This principle is immortalized in the following famous quote attributed to John von Neumann (by Enrico Fermi): "With four parameters I can fit an elephant, and with five I can make him wiggle his trunk" (Dyson, 2004).

Von Neumann's quote is most apt for David Marr's (1982) highest level of mathematical modeling – what he called the computational level (and what is often referred to in mathematical psychology as the descriptive level). When a parameter is added to a computational-level model, its sole purpose is almost always to fit more data. Therefore, it is almost always true that every new parameter greatly increases mathematical flexibility. However, as one moves down the Marr hierarchy, the quote becomes less and less relevant. At Marr's algorithmic level (called process models in mathematical psychology), new parameters are added to model some new (e.g., psychological) process. Modeling a new process will generally increase mathematical flexibility, but one new process could add several new parameters that work together. This trend continues at Marr's lowest, implementational level where the goal is to model the hardware that implements the algorithms. Now new parameters are added to model new structure and in many cases, a new process might require several new structures, each of which requires several new parameters to model.

For these reasons, implementational-level models gener-

ally have many parameters but are nevertheless mathematically inflexible. CCN models are at Marr's implementational level, and as expected, they tend to be very inflexible. The inflexibility is built in via the architectural and process constraints supplied by the relevant neuroscience literature. For example, consider a CCN model that includes cortical and striatal units. The equations describing each unit will be characterized by a number of free parameters (for details, see the section entitled "Models of Single Spiking Neurons") and there will be other parameters that describe the strength of the cortical-striatal synapses. But because the projection from cortex to striatum is excitatory and one way, changing the values of any of these parameters can only have a very limited effect on the behavior of the model. For every unit, there are two sets of parameter settings: those that allow the model to fire spikes and those for which the model never fires spikes. The latter settings are disallowed because they fail to produce neuron-like behavior. Any parameter combination in the former group of settings will cause similar behavior in the model – namely, any condition that causes cortical units to increase their firing rate will also cause striatal units to increase their firing rate. This is the only data profile that the model can produce, regardless of how many free parameters it contains, and regardless of the numerical values of those parameters.

This inflexibility is a strength of CCN models. If the results are qualitatively incompatible with the CCN predictions, then the architectural and process assumptions of the model are almost always incorrect. It is rarely the case that the mispredictions are caused by poor parameter estimation. As a result, invalid models are quickly rejected, hastening the scientific process.

6.2.3 Model Convergence

Another major advantage of the CCN approach is that two researchers independently modeling the same behavior are more likely to converge on highly similar models, and this convergence should cause the resulting models to have a permanence that is unusual with more traditional approaches. For example, the evidence is overwhelming that the hippocampus plays an important role in episodic memory consolidation. So any CCN model of episodic memory is likely to include a component that models some region(s) in hippocampus. Since the neuroanatomy of hippocampus is well understood, independently constructed CCN models of episodic memory therefore should include some highly similar components.

6.2.4 Ability to Unite Seemingly Disparate Fields

CCN modeling can sometimes uncover relationships among seemingly unrelated behaviors. This is especially likely when independent modeling efforts converge on common brain regions. For example, cognitive neuroscience models of information-integration category learning and implicit sequence learning independently identified similar cortical-striatal circuits (e.g., Ashby et al., 1998; Grafton, Hazeltine, & Ivry, 1995). This suggested that these two seemingly disparate behaviors might share some previously unknown deep functional similarity. Several studies explored this possibility. First, Willingham, Wells, Farrell, and Stemwedel (2000) showed that implicit motor sequence production is disrupted when the response key locations are switched, but not when the hands used to depress the keys are switched. Several studies subsequently showed that this same pattern of results holds for information-integration categorization (Ashby, Ell, & Waldron, 2003; Maddox, Bohil, & Ing, 2004; Maddox, Glass, O'Brien, Filoteo, & Ashby, 2010). Without linking categorization and sequence learning through their hypothesized underlying neural circuits, this dependence of information-integration categorization on response location learning would have been much more difficult to discover.

CCN models can also sometimes establish links between models that do not share any common brain regions. Researchers who build and test CCN models are typically interested in one subprocess more than others. For example, researchers interested in visual perception might couple a detailed model of some regions in visual cortex with an oversimplified model of response selection that produces a stylized output, whereas researchers interested in motor performance might couple a detailed model of primary motor cortex with an oversimplified model of visual perception that produces a stylized input to the motor module.

If different researchers each build valid CCN models of their module of interest, then it should be fairly straightforward to create a new model by linking the two separate CCN models together, and this new model should be consistent with all the behavioral and neuroscience data that are consistent with either model alone. Furthermore, by relying on neuroscience data, CCN modeling can even instruct us on *how* to connect the models together (e.g., by looking at which regions project to which). This ability to combine models of different processes is not trivial; even if each of two models work in their given domain, there is no guarantee that the combined model will behave as desired.

6.3 CCN Modeling Principles

In traditional cognitive-based mathematical modeling of behavior, the overriding criterion for establishing the validity of a model is goodness-of-fit to the behavioral data (usually penalized for model complexity; see, e.g., Pitt, Myung, & Zhang, 2002). Unfortunately, there are many examples where models making very different cognitive assumptions provide approximately equal levels of goodness-of-fit, so in many cognitive domains there are many competing mathematical models that make very different cognitive assump-

tions. In most cases, there is not much that can be done to resolve these problems. One solution is to invoke the *Principle of Correspondent Change* (Townsend & Ashby, 1983), which states that the correct model should account for behavioral changes across experimental conditions by only changing the values of parameters that correspond to the independent variables that were varied to create the various conditions. So for example, if the different conditions are identical except for the brightness of the stimulus, then to fit the resulting data the correct model should only need to change the value of sensory parameters (so decision parameters should be invariant across conditions). Although the Principle of Correspondent Change is rarely invoked, when used it can prove effective (e.g., Van Zandt, Colonius, & Proctor, 2000).

The Principle of Correspondent Change is an attempt to add extra constraints to computational- and algorithmic-level models. The constraints that define CCN can also be formalized as a set of principles (Ashby & Helie, 2011; Meeter, Jehee, & Murre, 2007; O'Reilly, 1998). This section describes three of the five principles that were proposed as constraints on CCN modeling by Ashby and Helie (2011). For more discussion, or an alternative list of principles, see Ashby and Helie (2011), Meeter et al. (2007), or O'Reilly (1998).

6.3.1 The Neuroscience Ideal. *A CCN model should not make any assumptions that are known to contradict the current neuroscience literature.*

This principle formalizes the CCN goal of building models that are constrained by existing neuroscience data. Note however, that the Neuroscience Ideal does not say that a CCN model must be compatible with all existing neuroscience data. A model is an abstraction, and therefore almost always is incomplete. The brain is immensely complex, and every CCN model must omit much of this complexity. The Neuroscience Ideal weighs these errors of omission much less heavily than errors of commission (Meeter et al., 2007). For instance, it is common in CCN modeling to omit connections between some brain areas that are known to be connected. This is done to keep the model simple and to focus on other connections that the model assumes are functionally important. A good fit suggests the missing connections might not be functionally important to the phenomenon under study. On the other hand, the Neuroscience Ideal disallows connections between brain areas that are known to not be connected or to creating an inhibitory connection between two regions when it is known that the real projection is excitatory.

Because of the great complexity of the human brain, every CCN modeling project must decide on an appropriate level of reductionism. The Simplicity Heuristic is a guide to solving this problem.

6.3.2 The Simplicity Heuristic. *No extra neuroscientific detail should be added to the model unless there are data to test this component of the model or past research has shown that the neuroscientific detail is a major contributor to the explanation.*

This heuristic is an application of Occam's razor. It is especially important because unlike cognitive models, with CCN models there will almost always be many extra neuroscientific details that one could add to an existing model. Doing so will increase the complexity of the model, the number of free parameters, and the computing time required for fitting. Unless there are data to test these extra components, it will be impossible to know whether these extra details were modeled correctly, and to what extent these untested details contributed to the model's success.

6.3.3 The Set-in-Stone Ideal. *Once set, the architecture of the network and the models of each individual unit should remain fixed throughout all applications.*

This could be considered a corollary to the Principle of Correspondent Change. Connections between brain regions do not change from task to task, nor does the qualitative nature via which a neuron responds to input. Thus, the model's analogues of these features should also not change when the empirical application changes. This ideal greatly reduces the mathematical flexibility of CCN models. For example, although a CCN model will initially have many unknown constants, most of these will be set by single-unit recording data and then, by the Set-in-Stone Ideal, they will remain invariant across all applications of the model.

6.4 Models of Single Spiking Neurons

The units that comprise the neural network in a CCN model should include more biological detail than the units used in traditional neural network theory (e.g., Haykin, 2009) or connectionism (Rumelhart & McClelland, 1986). CCN models follow two different general approaches to this problem. One approach builds networks from units that mimic spiking neurons, and another builds networks that model the firing rates of populations of neurons in different brain regions. This section describes spiking-neuron models and the next section describes firing-rate models.

The spiking-neuron models used in CCN originate in the classic Hodgkin-Huxley model (1952), which is a set of four coupled differential equations. One describes fast changes in intracellular voltage and three describe slow changes in various ion concentrations (i.e., Na+, K+, and Cl-). The model correctly accounts for action potentials (both the upstroke and downstroke), the refractory period, and subthreshold depolarizations that fail to produce a spike. From a CCN perspective, the model has several disadvantages. First, it was created to model voltage changes in the giant squid axon, rather than in mammalian neurons. Second, four differential equations must be solved numerically for every unit in the model. As a result, models that include many units could require prohibitive computing time. Third, for most CCN applications the Hodgkin-Huxley model violates the Simplicity Heuristic because rarely do such applications attempt to ac-

count for data that depend on intracellular concentrations of sodium, potassium, or chloride.

For these reasons, there have been a number of attempts to produce models with fewer equations that display as many of the desirable properties of the Hodgkin-Huxley model as possible. Some of these attempts are described in the following subsections.

6.4.1 The Leaky Integrate-and-Fire Model

The simplest model of a single unit that produces spiking behavior, and also the oldest (Lapique, 1907), is the leaky integrate-and-fire model (e.g., Koch, 1999). Suppose neuron B receives an excitatory projection from neuron A. Let $V_A(t)$ and $V_B(t)$ denote the intracellular voltages at time t in neurons A and B, respectively. Then the leaky integrate-and-fire model assumes that the rate of change of $V_B(t)$ is given by

$$\frac{dV_B(t)}{dt} = \alpha f\,[V_A(t)] + \beta - \gamma V_B(t), \qquad (1)$$

where α, β, and γ are constants. The function $f\,[V_A(t)]$ models temporal delays in the propagation of an action potential from the pre- to the postsynaptic unit. This function is described in detail in the subsection entitled "Modeling Synaptic Delays", but briefly it models temporal delays that occur when an action potential propagates down the axon in the presynaptic unit, and the temporal smearing that occurs during the chemical cascades initiated by this action potential in the synapse and in the dendrites of the postsynaptic unit. The parameter α is a measure of synaptic strength because the larger this value the greater the effect of an action potential in the presynaptic unit. In many applications, learning is modeled by assuming that α changes as a function of experience. The parameter β determines the spontaneous firing rate of unit B, and γ determines the rate at which charged ions leak out of the unit.

Equation (1) is a linear differential equation that does not produce spikes. Instead it predicts continuous and smooth changes in activation. To generate spikes from this model a threshold V_{peak} is set on $V_B(t)$. When $V_B(t)$ exceeds V_{peak} it is reset to V_{reset} and a spike is drawn by hand. An example of activation produced by this model is shown in Figure 1. The top panel shows the membrane potential predicted by the model when V_{peak} = -10 and V_{reset} = -50. The bottom panel adds hand-drawn spikes.

The leaky integrate-and-fire model is simple enough that it can be investigated analytically. In fact, in the absence of any input from the presynaptic neuron A, solving Eq. 1 shows that $V_B(t)$ is just an exponential function. For almost all other models however, analytic solutions are unavailable, so numerical solutions are required. In these cases, it can be difficult to predict how the activation will change if any of the numerical constants are changed or if the model is modified in any other way. One solution to this problem is to apply methods from nonlinear dynamics (e.g., Strogatz, 2014; Wiggins, 2003).

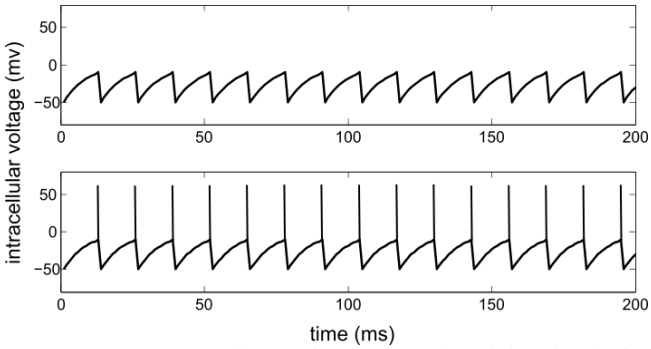

Figure 1. Top Panel. Activation produced by the leaky integrate-and-fire model (with $\beta = 1/60$, $\gamma = 7/60$, $V_{peak} = -10$, and $V_{reset} = -50$) . Bottom Panel. The same activation as in the top panel, except with spikes added by hand.

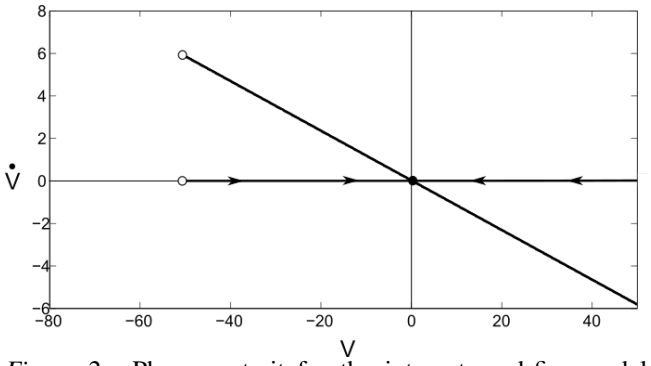

Figure 2. Phase portrait for the integrate-and-fire model (with $\beta = 1/60$, $\gamma = 7/60$, $V_{peak} = -10$, and $V_{reset} = -50$). The abscissa is intracellular voltage $V_B(t)$, whereas the ordinate is $\dot{V}_B(t) = \frac{dV_B(t)}{dt}$

.

For example, consider Figure 2, which shows a sort of phase portrait for the leaky integrate-and-fire model. The abscissa is intracellular voltage, and thus is the same as the ordinate in Figure 1. The Figure 2 ordinate denotes values of $\frac{dV_B(t)}{dt}$ [denoted there by $\dot{V}_B(t)$]. Thus, voltage increases for any value of V_B for which $\dot{V}_B(t) > 0$ and voltage decreases for any V_B for which $\dot{V}_B(t) < 0$. Note that voltage increases for any negative value of V_B (actually for any $V_B < 1/60$) and decreases when V_B is positive (i.e., when $V_B > 1/60$). When V_B is exactly equal to 1/60, the derivative is 0 and therefore, voltage will remain at this value unless or until some external input is added to the model. Thus, $V_B = 1/60$ is a stable fixed point of the model – or in other words, an attractor.

Figure 2 also shows that as voltage increases from the reset value (-50 mv), the derivative continuously decreases, so although voltage increases to the attractor, it does so in a slower and slower manner. In other words, the plot of voltage against time must be negatively accelerating – a property that

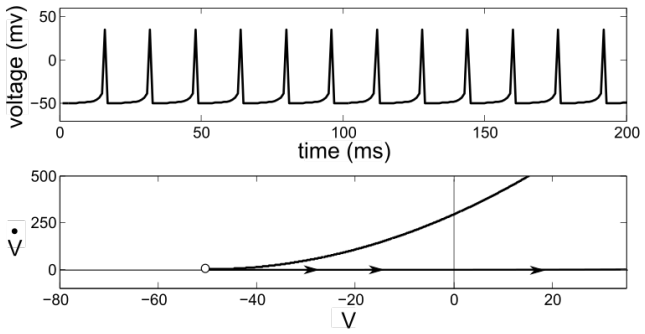

Figure 3. *Top Panel*. Typical spiking profile produced by the quadratic-integrate-and-fire model. *Bottom Panel*. Phase portrait for the quadratic-integrate-and-fire model. In both cases $\beta = 11.83$, $\gamma = .117$, $V_r = -60$, $V_t = -40$, $V_{peak} = 35$, and $V_{reset} = -50$.

is easily seen in Figure 1.

The fact that the leaky integrate-and-fire model does not naturally predict spiking is widely considered a weakness of the model (e.g., Izhikevich, 2007). Also, it does a relatively poor job of describing msec by msec changes in the membrane potential of real neurons and it is not flexible enough to model qualitative differences in the dynamics of different types of neurons. For these reasons, other single-equation models have been developed.

6.4.2 The Quadratic Integrate-and-Fire Model

Perhaps the most popular single-equation alternative to the leaky integrate-and-fire model replaces the linear decay term with a quadratic polynomial. The resulting model is known as the quadratic integrate-and-fire model (Ermentrout, 1996; Latham, Richmond, Nelson, & Nirenberg, 2000). For the scenario modeled in Eq. (1), the quadratic integrate-and-fire model assumes that the rate of change of $V_B(t)$ is given by

$$\frac{dV_B(t)}{dt} = \alpha f\left[V_A(t)\right] + \beta + \gamma\left[V_B(t) - V_r\right]\left[V_B(t) - V_t\right], \quad (2)$$

where α, β, and γ are constants, V_r is the resting membrane potential, V_t is the instantaneous threshold potential and, as before, the function $f[V_A(t)]$ models temporal delays in the propagation of an action potential from one neuron to another. Unlike the leaky integrate-and-fire model, Eq. (2) produces the upstroke of action potentials via its natrual dynamics, although it does not produce the downstroke. To create spikes, an extra voltage resetting step is required to generate the downstroke of the action potential – specifically, when $V_B(t)$ reaches V_{peak} it is reset to V_{reset}. Figure 3 shows an example of the spiking behavior produced by Eq. (2).

The bottom panel of Figure 3 shows the model's phase portrait. Note that the derivative $\dot{V}_B(t)$ is always positive, so in this model intracellular voltage can only rise. This is why the artificial voltage resetting mechanism is required. Also not that $\dot{V}_B(t)$ is positively accelerating. Thus, as the voltage increases, the magnitude of the increase becomes progressively greater. This is the property that allows the model to account for the upstroke of the action potential. Because of this property, the quadratic integrate-and-fire model is generally viewed as a superior alternative to the leaky integrate-and-fire model (Izhikevich, 2007).

6.4.3 The Izhikevich Model

Much more realistic behavior is possible if a second differential equation is added that models slow changes in ion concentrations. One of the first of these was the FitzHugh-Nagumo model, in which the rate of change in voltage (i.e., the derivative) is modeled as a cubic polynomial and slow changes in ion concentrations are modeled with a linear differential equation (FitzHugh, 1961; Nagumo, Arimoto, & Yoshizawa, 1962). Izhikevich (2003) proposed a similar model that replaces the cubic polynomial with the quadratic integrate-and-fire model. The Izhikevich (2003) model requires less computing time to evaluate than the FitzHugh-Nagumo model, has simpler dynamics, and can account for some qualitative firing phenomena that are outside the scope of the FitzHugh-Nagumo model (e.g., tonic and rebound bursting; Izhikevich, 2004). The Izhikevich (2003) model assumes

$$\begin{aligned}\frac{dV_B(t)}{dt} &= \alpha f\left[V_A(t)\right] + \beta + \gamma\left[V_B(t) - V_r\right]\left[V_B(t) - V_t\right] - \theta U_B(t),\\ \frac{dU_B(t)}{dt} &= \lambda\left[V_B(t) - V_r\right] - \omega U_B(t), \end{aligned} \quad (3)$$

where the quadratic integrate-and-fire model is as before and θ, λ, and ω are constants. In these equations $V_A(t)$ and $V_B(t)$ again denote intracellular voltages at time t and $U_B(t)$ is an abstract regulatory term that is meant to describe slow recovery in unit B after an action potential is generated. $U_B(t)$ could represent activation in the K+ current or inactivation in the Na+ current, or some combination of both. As before, when $V_B(t)$ reaches V_{peak} it is reset to V_{reset}. At the same time however, $U_B(t)$ is also reset to $U_B(t) + U_{reset}$).

The Eq. 3 model is highly flexible and produces some extremely realistic spiking behavior. Figure 4 shows examples of four qualitatively different kinds of dynamical behavior that can be produced from this model (from Izhikevich, 2003) when different numerical values are chosen for its parameters, and Izhikevich (2003) has identified at least 17 other types. Figure 4 shows predictions from the various models in the same patch-clamp type experiment where current is injected into the unit at the same time in each case. Especially when noise is added, many of the spike trains produced by the models are almost indistinguishable from single unit recordings collected from real neurons (for many exam-

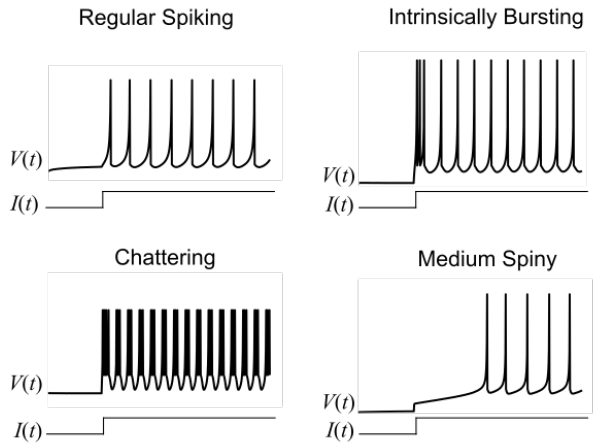

Figure 4. Spike trains produced by four different versions of the Izhikevich (2003) model in the same patch-clamp experiment, where the injected current is denoted by $I(t)$. (Regular spiking neuron: $\beta = .52, \gamma = .007, \theta = .01, \lambda = -.06, \omega = 0.03, V_r = -60, V_t = -40, V_{peak} = 35, V_{reset} = -50, U_{reset} = 100$. Intrinsically bursting neuron: $\beta = .52, \gamma = .012, \theta = .01, \lambda = .05, \omega = 0.01, V_r = -75, V_t = -45, V_{peak} = 50, V_{reset} = -56, U_{reset} = 130$. Chattering neuron: $\beta = 1.04, \gamma = .03, \theta = .02, \lambda = .09, \omega = 0.03, V_r = -60, V_t = -40, V_{peak} = 25, V_{reset} = -40, U_{reset} = 150$. Medium spiny neuron: $\beta = 2, \gamma = .02, \theta = .02, \lambda = -.2, \omega = 0.01, V_r = -80, V_t = -25, V_{peak} = 40, V_{reset} = -55, U_{reset} = 150$.)

ples, see Chapter 8, Izhikevich, 2007). One reasonable strategy, which follows from the Simplicity Heuristic, is to use the Izhikevich model for any units in the network for which single-unit recording data are available. If no such data are available then the simpler quadratic integrate-and-fire model could be used instead. Numerical solutions of Eqs. (1) – (3) are readily obtained using Euler's method. For example, Izhikevich (2007) provides Matlab code that solves Eqs. (3) using this approach.

The Izhikevich model has many free parameters that each affect the resulting dynamics of the model – oftentimes in interrelated ways. As a result, finding parameter estimates for a neuron type that has not been previously modeled can be a daunting task. Studying the phase portrait can greatly facilitate this process. For example, consider Figure 5, which illustrates the phase protrait of the regular spiking neuron. The top panel shows the spike train produced by the model in a patch-clamp experiment in which current is injected into the neuron beginning at time $t = 900$ ms and ending at time $t = 1200$ms. Notice that the neuron fires a burst to the injected current, and then immediately transitions to its slow spontaneous firing (with rate completely determined by the parameter β). The middle and bottom panels show phase portraits that describe the model's qualitative behavior. The middle panel shows the phase portrait while the current is

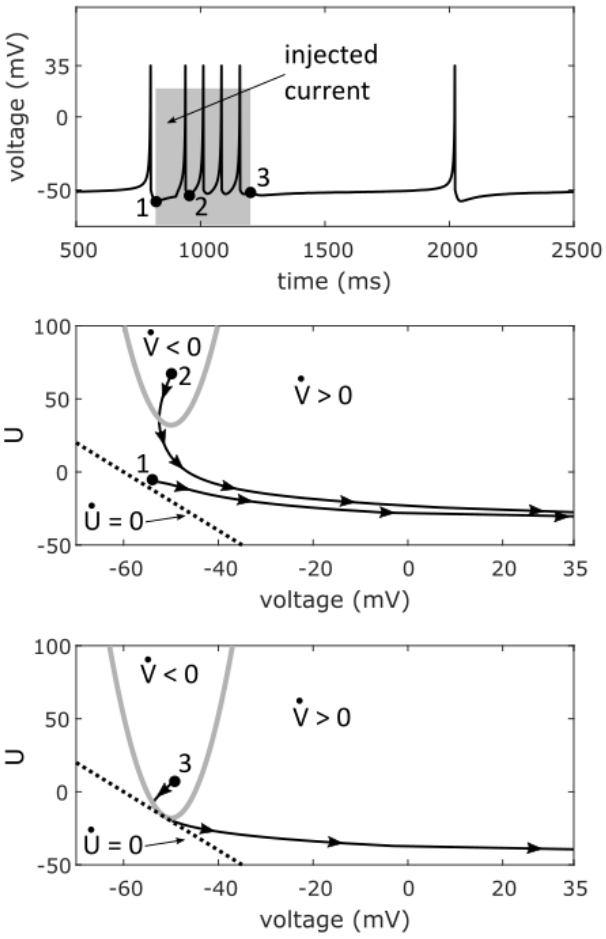

Figure 5. *Top Panel*. Typical spiking profile produced by the regular spiking neuron version of the Izhikevich (2003) model in a patch-clamp experiment where current is injected into the unit (denoted by the gray box). *Middle Panel*. Phase portrait during the time of the injected current. *Bottom Panel*. Phase portrait during the time after the injected current.

being injected and the bottom panel shows the phase portrait after the current has been turned off. In both figures the abscissa shows values of the intracellular voltage $V_B(t)$ and the ordinate shows values of $U_B(t)$. The gray quadratic curve, known as the voltage nullcline, denotes values of the (V_B, U_B) ordered pair for which $\dot{V}_B(t) = 0$. The derivative is negative for all points inside the parabola and positive for all outside points. The dotted black line is the U_B nullcline [i.e., the set of all (V_B, U_B) for which $\dot{U}_B(t) = 0$], with $\dot{U}_B(t) < 0$ for all points above the line and $\dot{U}_B(t) > 0$ for all points below.

The numbers, 1, 2, and 3, in Figure 5 identify time points.

Time 1 is at 900 ms – right when the current is first injected. Note that the voltage has just been reset because the model has just spontaneously spiked. Time 2 is just after the voltage was reset following the first spike produced during current injection, and time 3 coincides with the offset of the injected current. Note that at time point 1, $\dot{V}_B(t) > 0$, so the intracellular voltage $V_B(t)$ begins to increase. The further the trajectory moves away from the voltage nullcline, the greater the value of $\dot{V}_B(t)$ and the faster the voltage increase. When the threshold for another spike is reached (i.e., at $V_B(t) = V_{peak}$, where V_{peak} = 35 mV in this case), the voltage is reset to $V_B(t) = V_{reset}$ (-50 mV in Figure 5) and $U_B(t)$ is reset to its current value plus U_{reset} (i.e., 100). At time 2, note that $\dot{V}_B(t) < 0$, so right after the second spike, voltage decreases slightly (readily apparent in the top panel). But as the trajectory crosses the voltage nullcline, $\dot{V}_B(t)$ switches to positive and voltage begins to increase again, ever more rapidly until another spike is produced.

Note that the effect of the injected current is to raise the voltage nullcline. This moves more (V_B, U_B) points into the region for which $\dot{V}_B(t) > 0$, which increases the model's firing rate. Thus, manipulating any parameters that affect the width or location of the voltage nullcline will affect the overall firing rate of the model (i.e., γ, V_r, V_t, and β). The delay between spikes can be manipulated via the resetting parameters (V_{reset} and U_{reset}) and the parameters that control $U_B(t)$ (i.e., λ and ω). For example, to increase the delay, parameter values should be chosen that cause the trajectory to reset near the voltage nullcline. This will guarantee that $\dot{V}_B(t)$ is near zero, and therefore that voltage will initially change only slowly.

6.4.4 Modeling Synaptic Delays

Regardless of which spiking-neuron model is used, the free parameters that determine the dynamics of each unit should be set so that the behavior of the unit is as consistent as possible with what is known about the behavior of the real neurons the unit is meant to model. Then by the Set-in-Stone Ideal these parameter values should remain invariant across all applications of the model.

The integrate-and-fire model, the quadratic integrate-and-fire model, and the Izhikevich model all describe changes in membrane potential at one particular spatial location within a neuron. They do not describe the propagation of action potentials throughout the cell. Nor do they model delays that occur when an action potential is propagated across a synapse. Modeling these phenomena is considerably more complex.

The standard approach to modeling the propagation of action potentials within a neuron (e.g., down an axon) is called compartment modeling (e.g., Koch, 1999). As mentioned earlier, in this approach a neuron is modeled as a collection of cylinders and spheres, each of which is called a compartment. Separate partial differential equations are written that describe the propagation of the action potential within each compartment and all these equations are used to predict how an action potential propagates from a dendrite down to the end of an axon. The standard partial differential equation that describe propagation within each compartment is called the cable equation (e.g., Koch, 1999). This approach is widely used to account for detailed results of patch-clamp experiments in which current is injected at one location on the neuron and the results are recorded at various other locations.

Modeling synaptic events is potentially even more complex. For example, when an action potential reaches the terminal end of an axon: 1) synaptic vesicles open, 2) neurotransmitter is released, 3) the neurotransmitter diffuses across the synapse, 4) the neurotransmitter binds to postsynaptic receptors, and 5) the neurotransmitter-receptor complex either allows positively charged ions to flow directly into the neuron (in the case of ionotropic receptors) or else initiates a chemical cascade that indirectly causes the postsynaptic neuron to become depolarized (in the case of metabatropic receptors). All these processes cause temporal delay in the propagation of the action potential from the presynaptic to the postsynaptic neuron, and they also introduce a temporal smear. The action potential is a spike but its postsynaptic effects are not. Modeling any one of these processes can be challenging[1].

Models that describe how action potentials propagate down an axon and cross synapses must necessarily be considerably more complex than the one- and two-equation spiking neuron models considered in this chapter. The benefit of this extra complexity is that such models can make predictions at a more reductionistic level than say, the Izhikevich model. The Izhikevich model gives good accounts of spike trains, but is unable to account for data at lower levels (e.g., patch-clamp experiments that record from more than one site on the neuron; experiments that measure changes in ion concentrations over time). The cost of the extra complexity needed to make predictions at lower levels is a dramatic increase in computing time. One goal of CCN applications is to account for at least some behavior. Thus, CCN models must include multiple units in multiple brain regions. As a result, computing time is a serious consideration. The current state-of-the-art in CCN modeling is to account for any data between behavior (at the highest level) and spike trains (at the lowest level). In such applications, compartment models and models of synaptic transmission would be used only to predict the time-course of the postsynaptic effects of a spike in the presynaptic neuron. If this is the goal then we should seek simpler alternatives.

The problem is to model the temporal delays of spike

[1]Note that most applications of compartment modeling do not model any of these synaptic events. This is because the most common goal of compartment modeling is to account for results of single-neuron patch-clamp experiments.

propagation and the temporal smearing that occurs at the synapse in a simple way that can be combined with any of the one- or two-equation models of spiking considered above. A standard solution is to use the so-called alpha function (Rall, 1967). This is the function $f[V_A(t)]$ in Eqs. (1) – (3). The idea is that every time the presynaptic unit spikes, the following input is delivered to the postsynaptic unit (with spiking time $t = 0$):

$$f(t) = \frac{t}{\delta} \exp\left(\frac{\delta - t}{\delta}\right). \tag{4}$$

This function has a maximum value of 1.0 and it decays to .01 at $t = 7.64\delta$. Thus, δ can be chosen to model any desired temporal delay. If a second spike occurs before $f(t)$ decays to zero then a second alpha function is added to the residual $f(t)$ (again, with time of the second spike at $t = 0$).

6.4.5 Noise

In many applications it will be desirable to add noise to the models. There are two primary advantages to this. First, of course, human behavior is almost always probabilistic. Without noise the models will always produce the same result given the same initial conditions. So noise is necessary to account for probabilistic responding. Second, noise can drive the model out of a dangerous attractor state that can arise with any global learning rule.

Learning rules in neural network models can be classified as local or global. Local rules, like backpropagation, modify every synapse using a different error signal. In contrast, global learning rules use the same error signal at every synapse. The evidence is good that most of the brain uses global learning rules (with the possible exception of the cerebellum). For example, dopamine (DA), which is widely thought to serve as a training signal, is released in roughly equal amounts at all target synapses. One dangerous property of global learning is that it can lead to an attractor state in which response accuracy is constrained to remain at chance. Fortunately, adding noise to a model can break it free from this dangerous attractor state.

As an illustration of this problem, consider a simple two-stimulus, two-response task in which the network must learn to emit one response if stimulus A is presented and another response if stimulus B is presented. So after training, presentation of stimulus A should activate motor unit A more strongly than motor unit B. Initially, before training, the strength of the synapse between sensory unit A and the two motor units should be roughly equal. If two synaptic strengths are exactly equal, then the pre- and postsynaptic activations will be identical at both synapses (since the presynaptic activation is from the same cortical unit), and therefore any global learning algorithm will specify an equal amount of strengthening or weakening of both synapses on every trial, regardless of whether the response was correct or incorrect. Thus, if there is no noise then once the weights become equal they must remain equal for all time, thereby preventing the network from learning the desired associations. Adding noise to the postsynaptic activation breaks the model free from this attractor state. When noise is added, the postsynaptic activations at the two synapses will not be the same, even if the presynaptic activations and synaptic strengths are identical. As long as the postsynaptic activations are different, the change in synaptic strength will be different at the two synapses and learning can proceed.

Noise can be added to each of the models by adding a white noise term to each voltage equation. For example, the leaky integrate-and-fire model then becomes

$$\frac{dV_B(t)}{dt} = \alpha f\,[V_A(t)] + \beta - \gamma V_B(t) + \sigma\epsilon(t), \tag{5}$$

where σ is a constant and $\epsilon(t)$ is white noise. Note that except for the first, input term, Eq. (5) is exactly equivalent to an Ornstein-Uhlenbeck process. So whereas $\epsilon(t)$ and $\epsilon(t+\tau)$ are statistically independent for any value of τ, $V_B(t)$ and $V_B(t+\tau)$ are not independent, at least for reasonably small values of τ (because of the leak created by the $\gamma V_B(t)$ term). In real biological systems, inertia prevents physical changes large enough to guarantee that $\epsilon(t)$ and $\epsilon(t+\tau)$ will be independent for small values of τ. As a result, many researchers have proposed that the Ornstein-Uhlenbeck process is a better model of biological noise processes than white noise (e.g., Ricciardi & Sacerdote, 1979).

Adding a white noise term to each voltage equation converts the differential equation into a stochastic differential equation (e.g., Øksendal, 2003). In general, this complicates the process of deriving numerical predictions from the model. First, of course, adding noise to a voltage equation introduces variability to the spike times, which will generally cause variability in the predicted value of every dependent variable. This variability complicates model evaluation (for details see the section entitled "Parameter Estimation and Model Evaluation").

Second, the white noise term will defeat many numerical algorithms that are commonly used to solve differential equations. This is because white noise is not smooth and many differential equation solvers dynamically adjust the step size depending on the smoothness of the solution. In the presence of white noise, these algorithms keep shrinking the step size in an unsuccessful attempt to find a smooth solution, and often will eventually fail because the solution is not smooth even when the step size is as small as possible. On the other hand, simpler methods that use a fixed step size – such as Euler's method – work well in the presence of noise.

6.5 Firing-Rate Models

The human brain contains somewhere on the order of 10^{11} neurons. Although Izhekivich actually constructed a spiking-

neuron model with this many units[2], simulating any behavioral task with this complex a model is essentially impossible with today's technology. Furthermore, the brain is thought to exhibit considerable redundancy. For example, many neurons in the same cortical column or hypercolumn exhibit similar firing properties. For reasons such as these, an alternative approach to spiking-neuron models tries to model mean activity in large populations of neurons. The key variable is the instantaneous firing rate within each of these populations, and as a result, this class of models is known as firing-rate models.

Firing-rate models originated with the pioneering work of Wilson and Cowan (1972, 1973), who based their work on mean-field approaches from statistical mechanics. A complete review of this large field is beyond the scope of this chapter. For more details, see the excellent chapters by Ermentrout and Terman (2010, Chapter 11) or Dayan and Abbott (2001, Chapter 7).

Firing-rate models make many strong assumptions. In particular, they assume that all neurons within a population (e.g., a cortical column) are statistically identical and in close spatial proximity of each other. In addition, it is assumed that the neurons within a population are randomly interconnected and that these interconnections are dense enough so that any two neurons in the same population are connected – either directly or via interneurons. Furthermore, each population is assumed to include many neurons, and only two kinds of populations are allowed – one in which all neurons are excitatory and another in which all neurons are inhibitory. Given these assumptions, the mean instantaneous firing rate in any single population is defined as

$$R(t) = \lim_{\Delta T \to 0} \frac{1}{\Delta T} \left(\frac{\text{\# of spikes during } (t, t+\Delta T) \text{ in population}}{\text{\# of neurons in population}} \right). \tag{6}$$

Note that as $\Delta T \to 0$, each neuron in the population either fires once during the interval $(t, t + \Delta T)$ or not at all. Therefore $R(t)$ can also be interpreted as the proportion of active neurons at time t, and as a result $0 \le R(t) \le 1$.

As mentioned earlier, a popular method for modeling the postsynaptic effects of a presynaptic spike is via the alpha function described by Eq. (4). According to this model, every presynaptic spike generates a new alpha function postsynaptically. Firing-rate models follow a similar approach. Suppose the firing rate in a presynaptic population of excitatory neurons is $R_{\text{pre}}(t)$, and further suppose that the mean synaptic strength between all neurons in the pre- and postsynaptic populations is w. Then in firing-rate models, postsynaptic activation at time t, denoted by $I_{\text{post}}(t)$, is equal to the convolution of the presynaptic firing rate $R_{\text{pre}}(t)$ and the alpha function, weighted by mean synaptic strength:

$$I_{\text{post}}(t) = w \int_0^t f(t-s) R_{\text{pre}}(s) ds, \tag{7}$$

where $f(t)$ is the alpha function of Eq. (4).

This approach is easily extended to more complex architectures. For example, suppose some postsynaptic population receives input from M_{E} populations of excitatory neurons and M_{I} populations of inhibitory neurons. Let $w_{\text{E}i}$ denote the mean strength of all synapses of excitatory presynaptic population i neurons onto neurons in the postsynaptic population, and let $w_{\text{I}j}$ denote the mean strength of all synapses of inhibitory presynaptic population j neurons onto postsynaptic neurons. Then firing-rate models predict that the mean activation in the postsynaptic population equals:

$$I_{\text{post}}(t) = \sum_{i=1}^{M_{\text{E}}} w_{\text{E}i} \int_0^t f(t-s) R_{\text{E}i}(s) ds - \sum_{j=1}^{M_{\text{I}}} w_{\text{I}j} \int_0^t f(t-s) R_{\text{I}j}(s) ds, \tag{8}$$

where $R_{\text{E}i}(t)$ is the firing rate in excitatory presynaptic population i at time t and $R_{\text{I}j}(t)$ is the firing rate in inhibitory presynaptic population j.

The Eq. (4) alpha function models postsynaptic effects of a spike [i.e., $f(t)$ in Eqs. (7) and (8)] as a gamma function. In firing-rate models, other choices are also used. An especially popular choice – largely because of mathematical tractability – is the exponential function $f(t) = (1/\tau)\mathrm{e}^{-t/\tau}$. In this case, Eq. (8) reduces to

$$\tau \frac{dI_{\text{post}}(t)}{dt} = \sum_{i=1}^{M_{\text{E}}} w_{\text{E}i} R_{\text{E}i}(t) - \sum_{j=1}^{M_{\text{I}}} w_{\text{I}j} R_{\text{I}j}(t) - I_{\text{post}}(t) \tag{9}$$

Note the similarity of this equation to the leaky integrate-and-fire model described in Eq. (1). The main difference is that artificial spikes are generated in the leaky integrate-and-fire model but not in Eq. (9).

The last step is to convert the postsynaptic activation into postsynaptic firing rate. This is necessary because the activations predicted by Eqs. (8) and (9) are theoretically unbounded, whereas the firing rate $R(t)$ is constrained to the interval [0,1]. The standard approach is to assume that the postsynaptic firing rate equals

$$R_{\text{post}}(t) = F\left[I_{\text{post}}(t)\right], \tag{10}$$

where F is a monotonically increasing function known as the activation function. A common choice is to assume that F is a (sigmoidal) logistic function. According to this model

$$R_{\text{post}}(t) = \frac{1}{1 + \exp\left(-\frac{I_{\text{post}}(t) - \alpha}{\beta}\right)}, \tag{11}$$

[2]This unpublished simulation, which made no attempt to simulate behavior, is described on his website at http://www.izhikevich.org/human_brain_simulation/Blue_Brain.htm.

where α and β are constants. The constants α and β can be used to model a nonzero tonic firing rate and that presynaptic firing does not cause postsynaptic firing rate to saturate. When $\alpha = 0$, note that $R_{\text{post}}(t) = .5$ if the activation is 0 (i.e., if $I_{\text{post}}(t) = 0$). Thus, when $\alpha = 0$ the tonic firing rate is substantial (i..e, half the neurons in the population are active at any given time). Larger values of α cause the tonic firing rate to decrease. In contrast, increasing the value of β lowers the asymptotic firing rate produced by any given (constant) activation level.

As an example, consider the simple case where a single presynaptic population of excitatory neurons projects to a postsynaptic population. Figure 6 shows predictions of Eqs. (9) and (11) when the firing rate of the presynaptic population equals .8 during the time interval [100,300] and is zero at all other times. Several features of these predictions deserve comment. First, note that the spontaneous firing rate equals 0 in the presynaptic population and .2 in the postsynaptic population (because $\alpha > 0$) – an illustration that firing-rate models can account for any tonic firing rate. Second, note that the model makes the reasonable prediction that a sudden onset of firing in the presynaptic population causes the postsynaptic firing rate to increase gradually (i.e., exponentially), and similarly that a sudden drop in presynaptic firing causes a gradual decrease (again exponential) in postsynaptic firing rate.

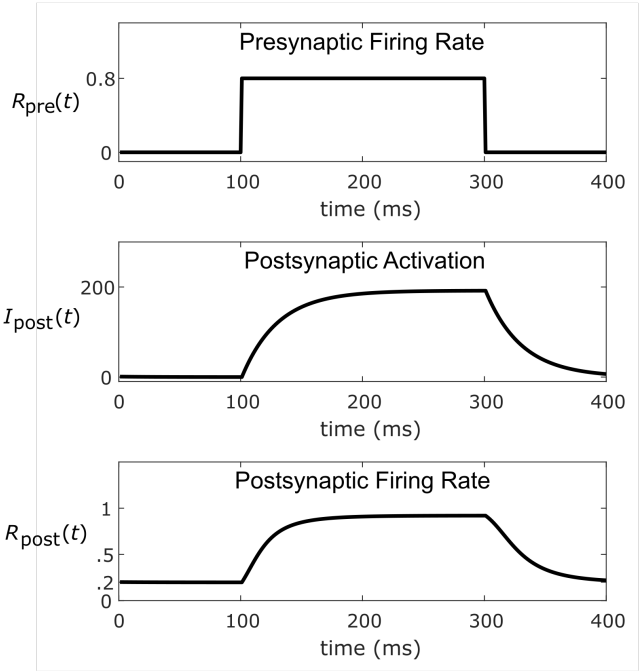

Figure 6. Predictions of a firing rate model for the case where an excitatory presynaptic population projects to a postsynaptic population. The top panel shows the firing rate of the presynaptic population. The middle panel shows the postsynaptic activation, and the bottom panel shows the postsynaptic firing rate (with $\tau = 30, w_{\text{E}j} = 240, \alpha = 70, \beta = 50$).

6.6 Learning

6.6.1 Synaptic Plasticity

The many different learning algorithms that can be found in the machine learning literature are often classified into three types. *Unsupervised learning* algorithms operate in the absence of any feedback or guidance. The most widely known version is Hebbian learning, in which all active synapses are strengthened, regardless of whether the response was rewarded or punished. *Reinforcement learning* algorithms depend on feedback but not on any guided instruction. They typically use a global learning algorithm that delivers the same feedback signal to every synapse in the network. *Supervised learning* algorithms depend on local feedback that supplies a unique training signal to every synapse. The most widely known examples include the delta rule and backpropagation.

The human brain exhibits many forms of neural plasticity that operate over a wide range of different time scales (for reviews, see e.g., Malenka & Siegelbaum, 2001; Stanton, Bramham, & Scharfman, 2006). The plasticity-related phenomena that are widely thought to be associated with learning and memory are long-term potentiation (LTP) and long-term depression (LTD) (e.g., Martin, Grimwood, & Morris, 2000). LTP and LTD refer to a long-lasting increase and decrease, respectively, in the efficacy of a synapse, which results from simultaneously stimulating the pre- and postsynaptic neurons. LTP and LTD have been closely studied in many different brain regions and in many different cell types. The conditions that promote LTP and LTD are qualitatively different in different regions, and it has been noted that these conditions closely match popular unsupervised-, reinforcement-, and supervised-learning algorithms. In particular Doya (2000) and others have noted that the rules that govern LTP and LTD match Hebbian learning in cortex and medial temporal lobe structures (including hippocampus; e.g., Feldman, 2009), reinforcement learning in the basal ganglia (and especially the striatum; e.g., Houk, Adams, & Barto, 1995), and supervised learning in the cerebellum.

The most common excitatory neurotransmitter in the brain is glutamate and virtually all long-range cortical projections are glutamatergic. For these reasons, it is fortunate that the most widely studied form of LTP occurs at glutamatergic synapses. Glutamate binds to a number of different types of receptors, but the most important for LTP are NMDA receptors. NMDA is an ionotropic receptor that is a channel for Na+ and Ca2+. It requires partial depolarization to become activated (because of an extracellular Mg2+ plug that prevents Na+ and Ca2+ from entering the cell during resting membrance potentials), and so it has a higher threshold for

activation than other non-NMDA glutamate receptors (e.g., AMPA).

NMDA-receptor activation initiates a number of chemical cascades that can affect synaptic plasticity. One of the most important and best understood is the pathway that phosphorylates calcium/calmodulin-dependent protein kinase II (CaMKII). When calcium enters the cell through the activated NMDA receptor, it binds to calmodulin and the calcium/calmodulin complex phosphorylates CaMKII. When fully phosphorylated, CaMKII initiates a variety of processes that eventually increase the efficacy of the synapse (e.g., Lisman, Schulman, & Cline, 2002). Dopamine (DA) plays an important role in these processes because if it is in the synapse within a second or so of the NMDA-receptor activation then it can potentiate the phosphorylating effects of calcium/calmodulin (via D1 receptor activation) and thereby potentiate synaptic efficacy (Yagishita et al., 2014).

A large literature shows that DA neurons in the ventral tegmental area (VTA) and substantia nigra pars compacta (SNpc) increase their firing above baseline following unexpected rewards (e.g., Hollerman & Schultz, 1998; Mirenowicz & Schultz, 1994; Schultz, 1998). Thus, this form of DA-enhanced LTP should be in effect following an unexpected reward in any brain region that is a target of VTA or SNpc DA neurons (and that expresses DA D1 receptors). This includes all of frontal cortex but not for example, visual or auditory cortex. In these regions however, there is evidence that acetylcholine may play a modulatory role similar to DA in LTP and LTD (e.g., Gu, 2003; McCoy, Huang, & Philpot, 2009). A variety of evidence suggests that the long-term efficacy of the synapse is weakened (i.e., LTD occurs) when presynaptic activation either fails to activate NMDA receptors, or else activates them only weakly (Bear & Linden, 2001; Kemp & Bashir, 2001).

Although the biochemistry of CaMKII-mediated synaptic plasticity is similar in all DA target regions, the functional role of this plasticity is qualitatively different in the striatum and frontal cortex. Within the striatum, DA is quickly cleared from synapses by DA active transporter (DAT) and, as a result, the temporal resolution of DA in the striatum is high enough for DA to serve as an effective reinforcement-learning signal. For example, if the first response in a training session is correct and the second response is an error then within the striatum, the elevated DA levels that result from the positive feedback on trial 1 should have decayed back to baseline levels by the time of the response on trial 2. Unlike the striatum however, DAT concentrations in frontal cortex are low (e.g., Seamans & Robbins, 2010). As a result, cortical DA levels change slowly. For example, the delivery of a single food pellet to a hungry rat increases DA levels in prefrontal cortex (PFC) above baseline for approximately 30 min (Feenstra & Botterblom, 1996). Thus, the first rewarded behavior in a training session is likely to cause frontal cortical DA levels to rise, and the absence of DAT will cause DA levels in frontal cortex to remain high throughout the training session. As a result, all synapses that are activated during the session are likely to be strengthened, regardless of whether the associated behavior is appropriate or not. Thus, although DA may facilitate LTP in frontal cortex, it appears to operate too slowly to serve as a frontal-cortical reinforcement training signal (Lapish, Kroener, Durstewitz, Lavin, & Seamans, 2007).

6.6.2 Models of Learning in the Striatum and Cortex

The structural changes that accompany LTP and LTD can be modeled in a variety of ways. One critical decision is whether to build a discrete-time or a continuous-time model. This choice largely depends on the nature of the data that the model will be tested against. If the data have a discrete trial-by-trial structure (i.e., the time is reset at the beginning of each trial), as is common in many cognitive-behavioral experiments, then a discrete-time model should be used because no data would exist to test the extra assumptions required of a continuous-time model. On the other hand, when modeling a continuous-time task (i.e., when the time is reset only once, typically at the beginning of the experiment), a continuous-time learning model is required. A cognitive example might be a sequence-learning task in which feedback is provided following each response and there is no pause between responses.

6.6.2.1 Discrete-time models of learning at synapses that lack fast DA reuptake. At synapses that lack fast DA reuptake, synaptic plasticity mimics Hebbian learning. In frontal cortex, for example, the first rewarded response should cause DA levels to rise above baseline and subsequent rewarded responses will cause DA to remain elevated for the duration of the training session. As a result, all active synapses will be strengthened, regardless of whether they received correct or error feedback. In this case, the key phenomena to model are that plasticity depends only on the product of pre- and postsynaptic activation. Strengthening of the synapse requires post-synaptic NMDA receptor activation. Activation below this threshold weakens the synapse.

Let $w_{\mathrm{A,B}}(n)$ denote the strength of the synapse on trial n between presynaptic unit A and postsynaptic unit B, and let $V_{\mathrm{J}}(t)$ denote the intracellular activation in unit J (J = A or B) at time t. The key variables to compute are the integrated alpha functions of units A and B. Suppose the time between stimulus presentation and response is T. Then define

$$I_{\mathrm{J}}(T) = \int_0^T f[V_{\mathrm{J}}(t)]\mathrm{dt} \tag{12}$$

for J = A or B. Note that $I_{\mathrm{J}}(T)$ describes the total postsynaptic effect of all spikes produced by unit J during the duration of the trial. Given these definitions, the following difference

equation can used to adjust the strength of the A/B synapse between trials n and $n+1$:

$$w_{\mathrm{A,B}}(n+1) = w_{\mathrm{A,B}}(n) + \alpha\, H\left[I_{\mathrm{B}}(t) - \theta_{\mathrm{NMDA}}\right] I_{\mathrm{A}}(t) \left\{1 - \mathrm{e}^{-\lambda[I_{\mathrm{B}}(t) - \theta_{\mathrm{NMDA}}]}\right\} [1 - w_{\mathrm{A,B}}(n)] - \beta\, H\left[\theta_{\mathrm{NMDA}} - I_{\mathrm{B}}(t)\right] I_{\mathrm{A}}(t)\, \mathrm{e}^{-\lambda[\theta_{\mathrm{NMDA}} - I_{\mathrm{B}}(t)]} w_{\mathrm{A,B}}(n) \qquad (13)$$

The function $H[g(x)]$ is the Heaviside function that equals 1 when $g(x) \geq 0$ and 0 when $g(x) \leq 0$. The constant θ_{NMDA} represents the threshold for NMDA-receptor activation. When postsynaptic activation is right at this threshold then the unit will produce a certain number of spikes during the trial duration T. Each spike generates an alpha function and θ_{NMDA} is theoretically equal to the integral of all these superimposed alpha functions. So the synaptic strengthening term is positive only on trials when the postsynaptic activation consistently exceeds the threshold for NMDA-receptor activation. When the synapse is strengthened, note that the amount of strengthening increases with the product of the presynaptic activation and an exponentially increasing function of the postsynaptic activation – similar to all other versions of Hebbian learning. The $[1 - w_{\mathrm{A,B}}(n)]$ term is a rate-limiting term that prevents $w_{\mathrm{A,B}}(n+1)$ from exceeding 1.0, and the constant λ scales the postsynaptic activation.

Most Hebbian learning rules do not include any mechanism to decrease synaptic strength. The last term in Eq. (13) is therefore unusual. First, note that this last term equals 0 except when postsynaptic activation is consistently below the NMDA-receptor threshold. Second, note that the exponential term reaches its maximum when postsynaptic activation is near the NMDA threshold and decreases as the postsynaptic activation gets smaller and smaller. This is consistent with the neurobiology. For example, in the absence of any postsynaptic activation we do not expect any synaptic plasticity. The $w_{\mathrm{A,B}}(n)$ at the end prevents $w_{\mathrm{A,B}}(n+1)$ from dropping below 0.

Figure 7 shows predicted changes in synaptic strength [i.e., $w_{\mathrm{A,B}}(n+1) - w_{\mathrm{A,B}}(n)$] for this model as a function of the magnitude of postsynaptic activation during both early [when $w_{\mathrm{A,B}}(n) = 0.2$] and late [when $w_{\mathrm{A,B}}(n) = 0.8$] learning.

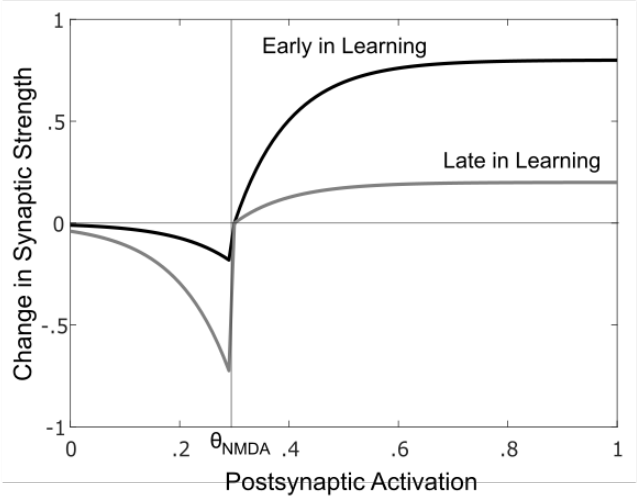

Figure 7. Change in synaptic strength predicted by the Hebbian learning model described in Eq. (13) as a function of amount of postsynaptic activation (here scaled from 0 to 1). Predictions are shown for early in learning [i.e., when $w_{A,B}(n) = 0.2$] and late in learning [i.e., when $w_{A,B}(n) = 0.8$].

6.6.2.2 Discrete-time models of learning at synapses with fast DA reuptake. In the striatum, DA reuptake is fast, so at cortical-striatal synapses LTP and LTD follow a form of reinforcement learning. One way to model synaptic plasticity at such synapses is as follows:

$$w_{\mathrm{A,B}}(n+1) = w_{\mathrm{A,B}}(n) + \alpha\, H\left[I_{\mathrm{B}}(t) - \theta_{\mathrm{NMDA}}\right] H[D(n) - D_{\mathrm{base}}] \times I_{\mathrm{A}}(t) \left\{1 - \mathrm{e}^{-\lambda[I_{\mathrm{B}}(t) - \theta_{\mathrm{NMDA}}]}\right\} [D(n) - D_{\mathrm{base}}][1 - w_{\mathrm{A,B}}(n)] - \beta\, H\left[I_{\mathrm{B}}(t) - \theta_{\mathrm{NMDA}}\right] H[D_{\mathrm{base}} - D(n)] \times I_{\mathrm{A}}(t) \left\{1 - \mathrm{e}^{-\lambda[I_{\mathrm{B}}(t) - \theta_{\mathrm{NMDA}}]}\right\} [D_{\mathrm{base}} - D(n)] w_{\mathrm{A,B}}(n) - \gamma\, H\left[\theta_{\mathrm{NMDA}} - I_{\mathrm{B}}(t)\right] I_{\mathrm{A}}(t)\, \mathrm{e}^{-[\theta_{\mathrm{NMDA}} - I_{\mathrm{B}}(t)]} w_{\mathrm{A,B}}(n), \qquad (14)$$

where $D(n)$ is the amount of DA released on trial n and D_{base} is the baseline DA level.

Note that the synaptic strengthening term requires two conditions – postsynaptic activation above the threshold for NMDA-receptor activation and DA above baseline. Once these conditions are met, synaptic strengthening is the same as in the Eq. (13) Hebbian-learning model. Two different conditions cause the synapse to be weakened. The second [the last γ term in Eq. (14)] is the same as in the Hebbian-learning model. The first (i.e., the β term) however, is unique to striatal-mediated reinforcement learning. Cortical-striatal synapses are weakened if postsynaptic activation is strong and DA is below baseline – a condition that would occur for example, on trials when feedback indicates the trial n response was incorrect.

Figure 8 shows predicted changes in synaptic strength [i.e., $w_{\mathrm{A,B}}(n+1) - w_{\mathrm{A,B}}(n)$] for this model as a function of the magnitude of postsynaptic activation, separately for early [when $w_{\mathrm{A,B}}(n) = 0.2$] and late [when $w_{\mathrm{A,B}}(n) = 0.8$] learning, and following correct and incorrect responses. Note that synaptic plasticity following correct (rewarded) responses is similar to plasticity in the Hebbian model (compare the top panel of Figure 8 with Figure 7). The only real difference is that learning is attenuated more during late learning in the reinforcement learning model. This is because DA fluctuations decrease as rewards become more predictable (more on this in the next subsection). Note that errors have a greater effect on synaptic plasticity late in learning. This is because errors are expected early in learning, so DA fluctuations are small. Late in learning however, when accuracy is high, er-

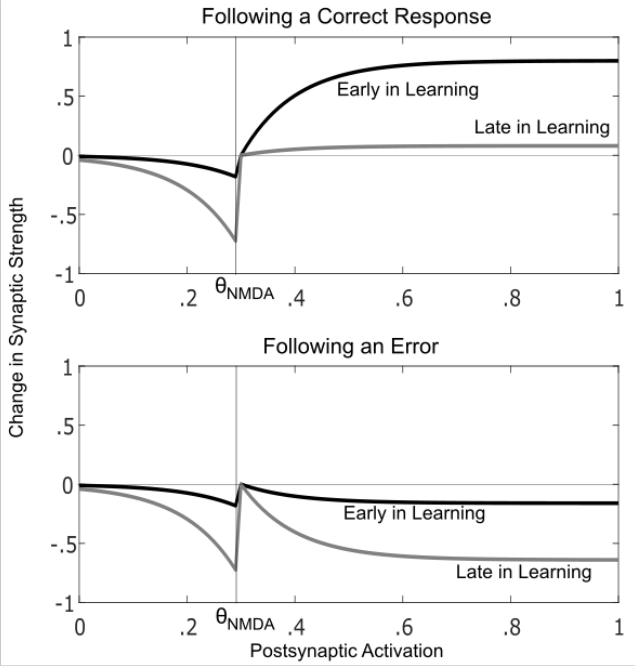

Figure 8. Change in synaptic strength predicted by the reinforcement learning model described in Eq. (14) as a function of amount of postsynaptic activation (here scaled from 0 to 1). Predictions are shown for early in learning [i.e., when $w_{A,B}(n) = 0.2$] and late in learning [i.e., when $w_{A,B}(n) = 0.8$], and following a correct response and an error. ($\alpha = 2, \beta = 4$, $\gamma = 1$).

rors are unexpected, which causes a large DA depression and therefore a large decrease in synaptic efficacy.

6.6.2.3 Modeling DA Release. The Eq. (14) model of reinforcement learning requires that we specify the amount of DA released on every trial in response to the feedback signal [the $D(n)$ term]. The more that DA increases above baseline (D_{base}), the greater the increase in synaptic strength, and the more it falls below baseline, the greater the decrease.

Although there are a number of powerful models of DA release, Eq. (14) requires only that we specify the amount of DA released to the feedback signal on each trial. The key empirical results are (e.g., Schultz, Dayan, & Montague, 1997; Tobler, Dickinson, & Schultz, 2003): (1) midbrain DA neurons fire tonically, and therefore have a nonzero baseline (i.e., spontaneous firing rate); (2) DA release increases above baseline following unexpected reward, and the more unexpected the reward the greater the release, and (3) DA release decreases below baseline following unexpected absence of reward, and the more unexpected the absence, the greater the decrease. One common interpretation of these results is that over a wide range, DA firing is proportional to the reward prediction error (RPE) – that is, to the difference between obtained reward and predicted reward. If we denote the obtained reward on trial n by R_n and the predicted reward by P_n, then the RPE on trial n is defined as:

$$RPE_n = R_n - P_n. \tag{15}$$

So positive prediction errors occur when the reward is better than expected, and negative prediction errors when the reward is worse than expected. Either signals that learning is incomplete.

A simple model of DA release can be built by specifying how to compute 1) obtained reward, 2) predicted reward, and 3) exactly how the amount of DA release is related to the RPE. A straightforward solution to these three problems is as follows (Ashby & Crossley, 2011). First, in tasks that provide positive feedback, negative feedback, or no feedback on every trial and where reward magnitude never varies, then a simple model can be used to compute obtained reward. Specifically, define the obtained reward R_n on trial n as +1 if correct or reward feedback is received, 0 in the absence of feedback, and -1 if error feedback is received.

Second, predicted reward can be computed using a simple average of past rewards, so long as the average employs temporal discounting in order to ensure that recent trials are weighted more heavily than earlier trials. Temporal discounting is critical to make the model sensitive to abrupt changes in reward probabilities. According to this approach, the predicted reward on trial $n + 1$ equals

$$P_{n+1} = \frac{1}{S_n} \sum_{i=1}^{n} \theta^{n-i} R_i, \tag{16}$$

where

$$S_n = \sum_{i=1}^{n} \theta^{i-1}.$$

So S_n is the sum of the weights on each R_i. Note that if $\theta = 1$ then $S_n = n$ and P_n equals the arithmetic mean of all past rewards. The more θ is reduced below 1, the greater the temporal discounting. Note that this model of predicted reward is stimulus and response dependent. Because predicted reward can vary greatly across stimuli and typically depends on what response is emitted, "trial $n + 1$" in Eq. (16) should be interpreted as the $(n+1)^{\text{th}}$ occurrence of this same stimulus and response.

Equation (16) is not the most convenient form for computing predicted reward – in part because it requires starting the sum from scratch on every trial. A more convenient form would be one where the current estimate of predicted reward is updated after each new reward is received. It turns out that Eq. (16) can be rewritten in such a manner as follows:

$$P_{n+1} = P_n + \frac{1}{S_n}(R_n - P_n). \tag{17}$$

Equations in this form are ubiquitous in the reinforcement learning literature (e.g., Sutton & Barto, 1998) – that is,

equations where the new estimate (e.g., P_{n+1}) is constructed by adjusting the old estimate (e.g., P_n) by a fractional amount of the prediction error (i.e., $R_n - P_n$). Models based on this form include the Rescorla-Wagner Model, temporal difference learning, SARSA, and Q-learning (e.g., see Sutton & Barto, 1998). Because of its great importance, a derivation of Eq. (17) is given next.

Derivation of Eq. (17). Equation (17) is derived from Eq. (16) as follows:

$$\begin{aligned}
P_{n+1} &= \frac{1}{S_n}\sum_{i=1}^{n}\theta^{n-i}R_i \\
&= \frac{1}{S_n}\left(R_n + \sum_{i=1}^{n-1}\theta^{n-i}R_i\right) \\
&= \frac{1}{S_n}\left(R_n + \theta\sum_{i=1}^{n-1}\theta^{n-1-i}R_i\right) \\
&= \frac{1}{S_n}\left(R_n + \frac{\theta S_{n-1}}{S_{n-1}}\sum_{i=1}^{n-1}\theta^{n-1-i}R_i\right) \\
&= \frac{1}{S_n}\left(R_n + \theta S_{n-1}P_n\right) \\
&= \frac{1}{S_n}R_n + \frac{\theta S_{n-1}}{S_n}P_n.
\end{aligned}$$

Next note that $\theta S_{n-1} = S_n - 1$. Therefore

$$\begin{aligned}
P_{n+1} &= \frac{1}{S_n}R_n + \frac{S_n - 1}{S_n}P_n \\
&= \frac{1}{S_n}R_n + P_n - \frac{1}{S_n}P_n,
\end{aligned}$$

from which Eq. (17) follows immediately. □

The final problem is to determine the amount of DA release associated with every possible value of RPE_n. A simple model was proposed by Ashby and Crossley (2011), who assumed that the amount of DA release is related to the RPE in a manner that is consistent with the data reported by Bayer and Glimcher (2005). Specifically, they assumed that

$$D(n) = \begin{cases} 1 & \text{if } RPE_n > 1 \\ .8RPE_n + .2 & \text{if } -.25 \le RPE_n \le 1 \\ 0 & \text{if } RPE_n < -.25. \end{cases} \qquad (18)$$

Note that the baseline DA level is .2 (i.e., when $RPE_n = 0$) and that DA levels increase linearly with the RPE. However, note also the asymmetry between DA increases and decreases. As is evident in the Bayer and Glimcher (2005) data, a negative RPE quickly causes DA levels to fall to zero, whereas there is a considerable range for DA levels to increase in response to positive RPEs[3].

6.6.2.4 Continuous-time models of Hebbian learning. All the learning models considered so far assume the data come from an experiment with a discrete-trial structure. In this case, synaptic strengths are updated off-line between trials. However, in continuous-time tasks the updating must be done in real time. This requires more detail than in the models we have so far considered. Not surprisingly, fewer continuous-time models of learning have been proposed.

Even so, one continuous-time model of Hebbian learning is widely used. This model was motivated by evidence that the magnitude and even the direction of plasticity at a synapse depends not only on the magnitude of the pre- and postsynaptic activations, but also on the timing – a phenomenon known as spike-timing-dependent plasticity (STDP). Considerable data show that if the postsynaptic neuron fires just after the presynaptic neuron then synaptic strengthening (i.e., LTP) occurs, whereas if the postsynaptic neuron fires first then the synapse is weakened (e.g., Bi & Poo, 2001; Sjöström, Rancz, Roth, & Häusser, 2008). Furthermore, the magnitude of both effects seems to fall off exponentially as the delay between the spikes in the pre- and postsynaptic neurons increases. Let T_{pre} and T_{post} denote the time at which the pre- and postsynaptic neurons fire, respectively. Then a popular model of STDP (e.g., Zhang, Tao, Holt, Harris, & Poo, 1998) assumes that the amount of change in the synaptic strength equals

$$\Delta = \begin{cases} e^{-\theta_+(T_{\text{post}} - T_{\text{pre}})}, & \text{if } T_{\text{post}} > T_{\text{pre}} \\ e^{\theta_-(T_{\text{post}} - T_{\text{pre}})}, & \text{if } T_{\text{post}} < T_{\text{pre}} \end{cases} \qquad (19)$$

Figure 9 shows an example of this function.

To implement this form of Hebbian learning, the strength of each synapse is updated according to Eq. (19) anytime the pre- and postsynaptic units both fire.

6.7 Testing CCN Models

CCN models can be tested against a wide variety of data, including data from single-unit recording, fMRI, and TMS experiments, as well as behavioral experiments with either healthy young adults or certain special neuropsychological patient groups (e.g., Parkinson's disease patients), who may or may not be operating under the influence of some drug (e.g., a DA agonist or antagonist). In most of these cases, some modeling interface is required to generate the relevant dependent measure from the neural activations that the models produce in each of their included brain regions. This sec-

[3]Bayer, Lau, and Glimcher (2007) subsequently reported that when the RPE is negative, DA firing remains below baseline for longer periods the more negative the RPE, suggesting that negative RPEs may be coded by a combination of firing rate and the duration of the pause in DA cell firing. This suggests that the dynamic range of positive and negative RPEs may be more balanced than assumed by the Eq. (18) model.

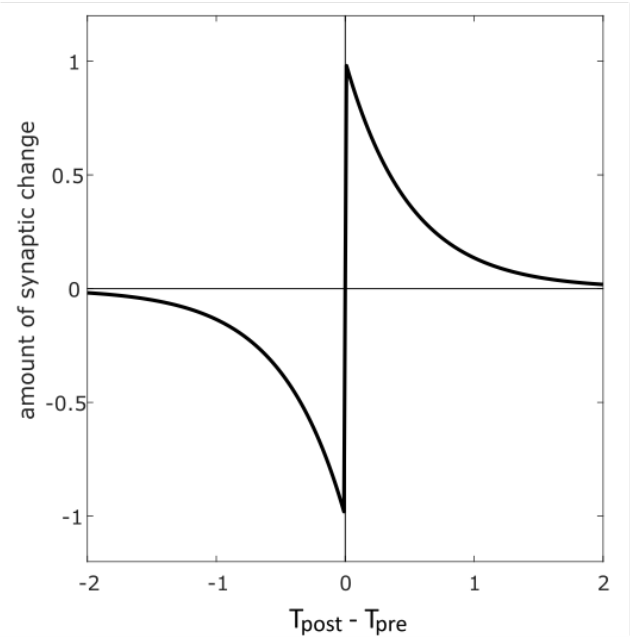

Figure 9. Amount of change in synaptic strength predicted by STDP as a function of the difference in time between firing in the postsynaptic neuron (i.e., T_{post}) and the presynaptic neuron (i.e., T_{pre}).

tion describes the most common of those interfaces and discusses some special issues that might arise during the modeling process.

6.7.1 Single-Unit Recording Data

One advantage that the spiking-neuron models hold over the firing-rate models is that only the former can be tested against single-unit firing data. A two-step process is recommended. During the first step, the models of each neuron type are separately fit to spike trains collected from single-neuron patch-clamp experiments. For example, if the Izhikevich spiking model is used then the end result of this first step will be numerical values for all constants in Eq. (3) that allow the model to provide good fits to the patch-clamp data. An example is shown in Figure 10.

Because of all the free parameters that are estimated, this step does not provide a test of the model. For example, because the patch-clamp data are from a single neuron, the network architecture of the model is irrelevant to goodness-of-fit. Even so, this step achieves two important goals. First, if successful, it guarantees that the units of the model have similar qualitative dynamics to the neurons they represent. And second, it fixes most of the free parameters of the model. By the Set-in-Stone Ideal, all free parameters that are fixed during this step must remain at these same fixed values during all future tests of the model. This is a major advantage of spiking-unit models over firing-rate models. In most cases,

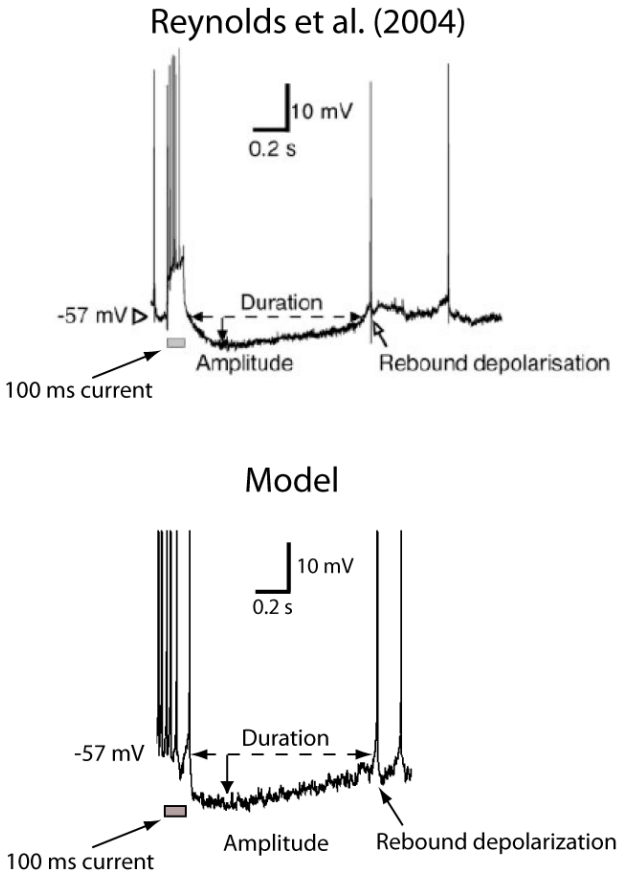

Figure 10. Patch-clamp recording data from the striatal TAN of a rat (top panel) and fits of a modified Izhikevich model under the same experimental conditions (from Ashby & Crossley, 2011).

all free parameters of firing-rate models must be estimated during the model-testing process, whereas most free parameters of spiking-unit models can be estimated during this first, preliminary step. Therefore, when fit to behavioral or fMRI data, spiking-unit models will typically require estimation of fewer free parameters than firing-rate models.

During the second step, the model is tested against single-unit recordings from the same neuron types as in the first step, except during *in vivo* recordings made while the animal is engaged in a behavior as similar as possible to the behavior that is the main focus of study. Since these recordings do depend on the network architecture, this step does provide a test of the model. The test is not parameter-free however, because some parameters will remain un-estimated after step 1. Mostly these will be synaptic strengths between different connected units in the model.

For example, Ashby and Crossley (2011) used this approach to test a CCN model of striatal function. They first

used patch-clamp data to build accurate Izhikevich models of two prominent striatal neuron types – namely, medium spiny neurons (96% of all striatal neurons) and TANs (i.e., tonically active neurons, which represent 2% of all striatal neurons; the TAN fit is shown in Figure 10). Next, with all these parameters held fixed, they showed that their network model accurately accounted for *in vivo* recordings from medium spiny neurons and TANs in a variety of different behavioral paradigms.

6.7.2 Behavioral Data

Spiking-unit and firing-rate networks produce activation in a distributed neural network, but without some added assumptions they produce no behavior. So to fit the models to behavioral data it is necessary to add some assumptions that describe how neural activation is related to behavior. In most cases, this process involves three steps. The first is to identify which brain region in the hypothesized network controls the behavioral response – that is, one must decide where to place the decision units. The second step is to decide, in each unit, what function of neural activity should drive the decision. For example, should the decision be based on the number of spikes, or the spiking rate, or perhaps on the integrated membrane potential? Finally, in tasks with multiple response alternatives, the third step is to decide how to resolve the competition among the various competing units in the critical brain region.

Step 1. What brain region controls behavior? The decision about where to place the decision units depends on one's knowledge of the task and the relevant neuroscience literature, and on one's modeling goals. In tasks that require finger or arm movements, typical choices would be the supplementary motor area, dorsal or ventral premotor cortex, or primary motor cortex. In contrast, if the task requires an eye-movement response then the critical area may be in the lateral intraparietal area, the supplementary eye fields, the frontal eye fields, or the superior colliculus. On the other hand, in many cases the goal may be to model cognition rather than the specific motor response that implements the outcome of the relevant cognitive processes. Ignoring motor processing simplifies the modeling because all areas downstream of the critical cognitive region can be omitted. Note that this strategy will underestimate RT since some key synapses will be omitted, but it might not affect accuracy predictions at all, especially in tasks where errors are due to cognitive failures, rather than to simple motor errors. For example, models of working memory typically assume that the key decision units are in PFC (e.g., Ashby, Ell, Valentin, & Casale, 2005; Frank, Loughry, & O'Reilly, 2001), since an extensive literature implicates the PFC as the most critical site for working memory. As a result, models of working memory often grossly oversimplify or omit altogether projections from PFC to premotor and motor cortices.

Step 2. What function of neural activity drives the decision? After the anatomical location of the decision units has been selected, the next step is to decide what function of activity in these units will initiate the behavior. With firing-rate models the obvious choice is to set a threshold on firing rate. When the threshold is crossed the behavior is initiated. With spiking-unit models, several choices are possible, but one especially appealing choice is to set a threshold on the integrated output alpha function:

$$\int_0^t f\,[V_B(x)]\,\mathrm{d}x, \tag{20}$$

where $f[\ ]$ is the alpha function defined in Eq. (4). The idea is to compute this integral continuously and initiate the behavior when the threshold is first exceeded. This decision variable has a number of attractive properties. Most importantly, it depends on the unit's output and because we expect the motor response to be driven by the output of the units in the decision region, the integrated alpha function is therefore as close to the behavior as possible without adding another downstream unit to the model (e.g., in contrast to a choice such as intracellular voltage, or even the number of spikes produced).

Step 3. How is a response selected when there are multiple alternatives? There has been considerable work on this problem in the field of neuroscience over the past decade or so. Especially illuminating have been studies in which single-unit recordings were made from putative decision neurons during a task in which an animal had to select among competing motor responses on each trial (for reviews, see e.g., Bogacz, Wagenmakers, Forstmann, & Nieuwenhuis, 2010; Rangel & Hare, 2010; Wang, 2008). For example, in an early and influential study, Shadlen and Newsome (2001) reported that neurons in the lateral intraparietal area reliably predicted the eye-movement responses of monkeys in a task that required the animals to determine the direction of motion of random dot patterns. Furthermore, these neurons displayed the push-pull profile that one might expect from a classic diffusion process – that is, neurons that predicted a movement of the eyes to the right increased their firing rate when the correct response to the stimulus was a rightward movement and decreased their firing rate when the stimulus signaled a leftward movement. The formal correspondence between these properties and the diffusion process was quickly noted (e.g., Smith & Ratcliff, 2004).

Of course, generalizing the diffusion model to more than two alternatives is not straightforward, but it is well known that an accumulator or race model with lateral inhibition among the channels mimics a diffusion process (Bogacz, Usher, Zhang, & McClelland, 2007; Usher & McClelland, 2001). Thus, in tasks with more than two response alternatives, a sound yet reasonably simple solution is to set a criterion on each decision unit and allow the first unit that

crosses this threshold to control the response, but also to build in lateral inhibition among all decision units (McMillen & Holmes, 2006; Usher & McClelland, 2001).

For example, suppose each of M decision units is modeled via the quadratic integrate-and-fire model described by Eq. (2). Suppose the J^{th} of these units receives input from m units in an earlier layer (or brain structure). Then we can model the lateral inhibition among output units on unit J via:

$$\frac{dV_{\mathrm{J}}(t)}{dt} = \alpha \sum_{i}^{m} f\left[V_i(t)\right] + \beta + \gamma \left[V_{\mathrm{B}}(t) - V_{\mathrm{r}}\right]\left[V_{\mathrm{B}}(t) - V_{\mathrm{t}}\right] - \omega \sum_{I \neq J}^{M} f[V_I(t)] \qquad (21)$$

The last term is a standard model of lateral inhibition (e.g., Usher & McClelland, 2001). Note that this model assumes that the total amount of lateral inhibition on unit J is an increasing function of the total amount of activation in all output units.

6.7.3 FMRI Data

One of the great advantages of CCN models over traditional cognitive models is that they can be tested against fMRI and other neuroscience data. CCN models predict changes in neural activation in a variety of different brain regions and fMRI records an indirect measure of neural activation. Thus, fMRI provides a natural platform from which to test CCN models. Nevertheless, some challenges must be overcome to take full advantage of fMRI methodology. First, CCN models make direct predictions about neural activation, but they do not make direct predictions about the blood-oxygen-level-dependent (BOLD) signal that is most commonly measured in fMRI experiments. Thus, the first problem is to generate predicted BOLD responses from the model's predicted neural activations. Second, CCN models make anatomic predictions about where the task-related changes in neural activation should be found, but they typically do not make predictions that are specific enough to identify a small set of voxels in the region of interest (ROI) that could be used to test the model. For example, a model might specify that during a certain period of a working memory task, a specific type of activation should occur in dorsolateral PFC (dlPFC). However, such a model would generally not predict that every voxel in dlPFC would show this activation pattern – only that some would. So, a second significant problem that must be solved is to identify exactly which voxels within dlPFC should be used to test the model. Finally, a third problem is to compare the observed and predicted BOLD responses in the selected voxels and to decide on the basis of this comparison whether the model succeeds or fails at accounting for the results of the experiment.

Although a variety of similar solutions to these problems have been proposed (Ashby & Waldschmidt, 2008), within the past few years consensus has settled on an approach, called model-based fMRI, that fully exploits current fMRI data-analysis software packages (O'Doherty, Hampton, & Kim, 2007). The basic idea is to first fit the model to the behavioral data collected during the scanning session separately for each participant. Next, parameter estimates from the model fits are used to generate predicted neural activations that are unique for every participant. The third step is to generate a predicted BOLD response from each brain region in the model, and then to correlate these predictions with observed BOLD responses from every voxel in the brain. These two steps can be performed by any of the popular fMRI data-analysis software packages. Finally, all the resulting correlations are assessed for statistical significance.

Model-based fMRI can be used to account for individual differences in fMRI data, but if the computational model is good, it can also be used to identify brain regions that respond selectively to components or sub-processes of the task. In particular, if the model has different parameters that describe different perceptual or cognitive processes that are presumed to mediate the behavior under study, then different regressors can be created that make specific predictions about each of these processes. For example, O'Doherty et al. (2004) used this approach to identify separate brain regions associated with the actor versus the critic in actor-critic models of reinforcement learning.

Steps 1 & 2. Generating predicted neural activations The first step is to fit the model separately to the behavioral data collected from each participant during the scanning session. Details on this process are given in the section entitled "Parameter Estimation and Model Evaluation." After this fitting process is complete, each participant will be characterized by a unique set of parameter estimates. The second step is to use these estimates to generate predicted neural activations for each brain region identified by the model. The critical issue to consider here is spatial resolution. A typical voxel size in fMRI is 2^3 to 3^3 mm^3. So the goal of this step should be to produce the model's best estimate of total neural activation in regions of about this size. This typically means that activations from all neurons within the same specified brain region should be added together.

Logothetis and colleagues reported evidence that the BOLD response is more closely related to local field potentials than to the spiking output of individual neurons (Logothetis, 2003; Logothetis, Pauls, Augath, Trinath, & Oeltermann, 2001). Local field potentials integrate the field potentials produced by small populations of cells over a submillimeter range, and they vary continuously over time. So if the CCN model is constructed from spiking units, the spike trains produced by the model must be converted to local field potentials. This can be done by lowpass filtering each spike. Fortunately, this is exactly the operation performed by the alpha function. So with spiking-unit models, after the scanning

task is simulated (separately for each participant) every spike within each brain region is used to trigger an alpha function and these are all added together to mimic the spatial summation that occurs during fMRI. These summed alpha functions represent the predicted local field potentials during the scanning session with the corresponding brain region. Predicted activation in firing-rate models is already temporally smoothed, so with firing-rate models no extra low-pass filtering is needed.

Step 3. Generating predicted BOLD responses The fMRI BOLD response increases with the amount of oxygenated hemoglobin in a voxel relative to the amount of deoxygenated hemoglobin (Ogawa, Lee, Kay, & Tank, 1990). In comparison with the neural activation that presumably drives it, the BOLD response is highly sluggish, reaching a peak around 6 sec after the neural activation that induced it, and slowly decaying back to baseline 20–25 sec later. Almost all current applications of fMRI assume that the transformation from neural activation to BOLD response can be modeled as a linear, time-invariant system (e.g., Boynton, Engel, Glover, & Heeger, 1996).

In the linear systems approach, one can conceive of the vascular system that responds to a sudden oxygen debt as a black box in which the input is neural activation, and the output is the BOLD response. If the system is linear and time invariant, then it is well known that the BOLD response at time t, denoted by $B(t)$, to any neural activation $N(t)$ can be written as

$$B(t) = \int_0^t N(x)h(t-x)\mathrm{d}x. \tag{22}$$

Equation (22) is the well-known convolution integral that completely characterizes the behavior of any linear, time-invariant system (see, e.g., Chen, 1970). The function $h(t)$ is traditionally called the impulse response function because it describes the response of the system to an input that is a perfect impulse. In the fMRI literature, however, $h(t)$ is known as the hemodynamic response function, often abbreviated as hrf. The hrf is the hypothetical BOLD response to an idealized impulse of neural activation, typically peaking at 6 sec and lasting for 30 sec or so. Thus, predicted BOLD responses can be generated from the model by numerically convolving the predicted neural activations with a suitable model of the hrf. Popular fMRI data analysis software packages such as SPM and FSL will perform the numerical convolution. All the user needs to do is supply a vector that contains the predicted neural activation at each TR of the scanning session and specify a functional form for the hrf. Many alternative models of the hrf have been proposed (e.g., a common choice is a gamma function), and the packages allow the user considerable flexibility with respect to this choice. For a complete description of this entire process, see Ashby (2011).

Step 4. Comparing predicted and observed BOLD responses Once predicted BOLD responses have been computed, the next step is to correlate these predictions with the observed BOLD responses in every voxel. All popular fMRI software packages routinely compute these correlations by using the general linear model (GLM) of statistics (see Ashby, 2011, for details). The standard analysis converts each correlation coefficient to the z- or t-statistic associated with the null hypothesis that the correlation is zero. The result is a z-statistic (for example) in every voxel in the brain, which collectively are known as a statistical parametric map (SPM).

Step 5. Assessing statistical significance The final step is to make a statistical significance decision about every z-statistic in the SPM. Of course with only one such decision the solution to this problem is taught in every introductory statistics course, but spatial resolution is high enough with modern imaging equipment that an adult human brain might be characterized by several hundred thousand voxels, and therefore several hundred thousand simultaneous significance decisions are required. To complicate matters further, spatial correlations guarantee that the separate statistics are not independent. Although there is no optimal solution to this problem, many different alternative methods have been proposed for correcting for this huge number of multiple comparisons. Current software packages allow the user to choose among many of these alternative solutions (again, see Ashby, 2011, for details).

This correlational analysis can be used in either a confirmatory or exploratory manner. The confirmatory analysis is to check whether significant correlations appear in the brain regions predicted by the model. Most CCN models will predict different neural activations in each brain region included in the model. The strongest possible confirmatory result would be that voxels in which the observed BOLD response is significantly correlated with the predicted BOLD response in one of these regions appear in that region but not in any of the other regions in the model, and that a similar unique confirmation is found for every hypothesized brain region. So for a model that includes brain regions A, B, and C, voxels where the BOLD response correlates with the predicted BOLD response in region A are found in region A, but not in regions B or C, and voxels where the BOLD response correlates with the predicted BOLD response in region B are found in region B, but not in regions A or C (and similarly for region C).

The exploratory analysis is to identify brain regions that are not in the model in which the observed BOLD response is nevertheless correlated with the predicted BOLD response for some region in the model. Such correlations could exist for a number of reasons, including because the model is incomplete.

6.7.4 TMS Data

TMS uses a small device to direct focused electromagnetic pulses through the skull of human participants. The stimulation typically targets a specific cortical site that was previously identified using high-resolution MRI. Many studies have used TMS to investigate the causal role of some specific cortical region in a particular cognitive process or behavior (e.g., Sandrini, Umiltà, & Rusconi, 2011).

Most current TMS studies use theta-burst stimulation, with typical protocols delivering a burst of 3 pulses at 50 Hz (i.e., 20 ms between each pair of pulses) and then repeating this pattern for as long as several minutes. Theta-burst TMS induces electrophysiological changes in the targeted site that last up to 60 min, and behavioral changes in tasks that depend on that site for a similar time period (Huang, Edwards, Rounis, Bhatia, & Rothwell, 2005). The current thinking is that theta-burst TMS induces long-term changes in synaptic plasticity within targeted regions by altering the pattern of Ca^{2+} influx through post-synaptic NMDA receptors (Huang, Rothwell, Chen, Lu, & Chuang, 2011). Whether synaptic strengthening or weakening are potentiated depends on the exact stimulation protocol. For example, a continuous 40 sec train of 3-pulse bursts has a long-term inhibitory effect, whereas a 2 sec train of 3-pulse bursts that is repeated every 10 sec for 3 min causes a long-term facilitatory effect. Huang et al. (2011) developed a mathematical model that accurately predicts the long-term effects of a wide variety of different theta-burst protocols.

In most TMS experiments the data of primary interest are behavioral (i.e., RTs and accuracies). The goal is typically to investigate how the TMS affects these behavioral measures. CCN models that assign a functional role to the cortical region targeted by the TMS can be tested against the resulting data. A particularly simple way to model the effects of theta-burst TMS within the CCN framework that is nevertheless consistent with the Huang et al. (2011) theory is to assume that TMS changes θ_{NMDA} in Eqs. (13) and (14) – that is, it changes the threshold for NMDA-receptor activation. Increasing θ_{NMDA} simulates a reduction in Ca^{2+} influx and will cause more synaptic weakening and less strengthening. In contrast, decreasing θ_{NMDA} simulates an increase in Ca^{2+} influx and causes less synaptic weakening and more strengthening. For example, Helie, Roeder, Vucovich, Rünger, and Ashby (2015) used this approach to show that their proposed CCN model of automatic sequence production successfully accounted for the interfering effects of TMS to the supplementary motor area on the RT speed-up that normally occurs during sequence learning (e.g., where the TMS data were reported by Verwey, Lammens, & van Honk, 2002).

6.7.5 Pharmacological and Neuropsychological Patient Data

Many CCN models also make predictions about how performance should change in relevant tasks when participants perform the task under the influence of certain medications or drugs, or when the participants are from some special neuropsychological population. For example, because the reinforcement learning model described in Eq. (14) includes a term that depends on the amount of DA released on each trial, any CCN model that uses this reinforcement learning algorithm should make specific predictions about how performance should be affected by any drug or neuropsychological condition that alters brain DA levels. This would include DA agonists and antagonists, and neuropsychological conditions such as Parkinson's disease. For example, Hélie, Paul, and Ashby (2012a, 2012b) used this approach to account for a variety of cognitive deficits that occur during Parkinson's disease and to account for the beneficial effects of positive mood on rule-based category learning.

6.8 Parameter Estimation and Model Evaluation

It is almost always impossible to derive predictions from CCN models analytically. Even the simplest CCN models are typically described by a rather large set of simultaneous nonlinear differential equations. At best these can be solved numerically. However, most CCN models will include noise terms, in which case the differential equations become stochastic. In this large majority of cases, Monte Carlo simulation is almost always necessary.

Finding best-fitting values of the parameters is a notoriously difficult problem when the model predictions are noisy and require simulation. The typical approach is to simulate many independent and identical subjects and then treat the mean of all these simulations as the model predictions. This can be time consuming and of course the resulting predictions will still be noisy and therefore standard minimization algorithms are not appropriate. Fortunately, genetic algorithms (e.g., Haupt & Haupt, 2004) can often be used successfully. For example, Cantwell, Crossley, and Ashby (2015) estimated parameters of a spiking-unit CCN model using particle swarm optimization (Clerc, 2012), which creates a population of potential solutions (the "particles") and then iteratively moves these particles in parameter space according to both their historically best position, and the best-known position of their neighborhood. Due to the stochastic nature of the models, the "function" to be optimized (e.g., sum of squared errors) is not strictly a function at all. Hence, particle swarm optimization, which makes very few assumptions about the form of the problem, is an appropriate tool where traditional optimization routines will fail. After parameter estimation was complete, Cantwell et al. (2015) ran an additional 100 simulations with the best-fitting parameter

values and the model predictions were computed by taking the mean across all 100 simulations.

The inflexibility of CCN models (i.e., see the section entitled "Advantages of CCN Modeling") eases the parameter estimation process. Small changes in almost any parameter usually cause only a negligible change in the model's predictions, and therefore small errors in the parameter estimation process will generally have little or no effect on any conclusions that are drawn about the empirical validity of the model. For example, following a crude parameter estimation process, Ashby and Crossley (2011) implemented a sensitivity analysis in which the most important parameters in their spiking-unit CCN model were successively changed by -1%, -10%, +1%, and +10%. After each change, the behavior of the model was simulated under the same experimental conditions that were used to generate the data the model was tested against. Next, after each new simulation, the correlation was computed between the predictions of the best-fitting model and the predictions generated from the new version of the model. In all except one case, these correlations exceeded .99, suggesting that the model makes the same qualitative predictions for a wide range of each of its parameters. The only exception occurred for a +10% increase in a response threshold parameter [i.e., the threshold mentioned in the discussion of Eq. (20)]. In this case, the correlation was .74. Importantly, however, even in this case, the perturbed model predicted the same qualitative pattern to the data as the best-fitting version of the model.

This inflexibility also means that parameter space partitioning (PSP; Pitt, Kim, Navarro, & Myung, 2006) will often allow stronger inferences when applied to CCN models than when applied to traditional process-level models. PSP determines what different kinds of qualitative data patterns a model can predict by systematically exploring the model's entire parameter space. Specifically, it uses an efficient Markov chain Monte Carlo search algorithm to compute the volume of parameter space over which the model can account for each different qualitative data pattern.

PSP is an especially effective method for rejecting a model because if observed data show a particular qualitative pattern and the PSP indicates that the volume of parameter space where the model can mimic this pattern is 0, then the model can be rejected. Of course, flexible models can fit more different types of data patterns, so the more flexible the model the less that can be learned from PSP. With CCN models however, PSP can lead to some strong inferences. For example, Valentin, Maddox, and Ashby (2016) used PSP to show that CCN models based on current theories of DA release are incompatible with observed effects of aggregate feedback on procedural learning. Similarly, Paul and Ashby (2013) used PSP to show that a large class of CCN models could only account for empirical interactions between declarative and procedural memory systems by predicting the existence of an as-yet undiscovered neuroanatomical projection from cortex to the striatum.

6.9 Conclusions

The birth of mathematical psychology did not signal the advent of mathematical modeling in psychology. For example, Fechner, Thurstone, and Hull all incorporated heavy doses of modeling into their research programs. Even so, these were all descriptive models, or in the language of Marr (1982), computational-level models. The birth of mathematical psychology could be seen as the beginning of process or algorithmic-level modeling in psychology. Stimulus sampling theory (Estes, 1950) sparked a revolution – not because it was the first mathematical model of learning (e.g., see Hull, 1943) – but at least in part because it was the first algorithmic-level model. Now, more than a half century later, it may be time to move to the next level of modeling – namely the implementational level. CCN represents a serious attempt to take this next step.

CCN presents new challenges to mathematical psychology. Some basic knowledge of neuroscience is required and parameter estimation tends to be more difficult than with many traditional process models. Even so, the potential benefits are significant. These include 1) model convergence, because different researchers must respect similar neuroscience constraints, 2) faster rejection of poor models, because of the mathematical inflexibility of CCN models, 3) the ability to test models against a much wider spectrum of data types, and 4) the potential to unite disparate fields when common brain regions are implicated in seemingly unrelated behaviors.

Author Notes

Preparation of this chapter was supported by National Institutes of Health Grant 2R01MH063760. Correspondence concerning this article should be addressed to the author at fgashby@ucsb.edu.

References

Anderson, J. R., Fincham, J. M., Qin, Y., & Stocco, A. (2008). A central circuit of the mind. *Trends in Cognitive Sciences*, *12*(4), 136–143.

Ashby, F. G. (2011). *Statistical analysis of fmri data.* Cambridge, MA: MIT Press.

Ashby, F. G., Alfonso-Reese, L. A., Turken, A. U., & Waldron, E. M. (1998). A neuropsychological theory of multiple systems in category learning. *Psychological Review*, *105*(3), 442-481.

Ashby, F. G., & Crossley, M. J. (2011). A computational model of how cholinergic interneurons protect striatal-dependent learning. *Journal of Cognitive Neuroscience*, *23*(6), 1549-1566.

Ashby, F. G., Ell, S. W., Valentin, V. V., & Casale, M. B. (2005). FROST: A distributed neurocomputational model of working memory maintenance. *Journal of Cognitive Neuroscience*, *17*(11), 1728–1743.

Ashby, F. G., Ell, S. W., & Waldron, E. M. (2003). Procedural learning in perceptual categorization. *Memory & Cognition*, *31*(7), 1114–1125.

Ashby, F. G., & Helie, S. (2011). A tutorial on computational cognitive neuroscience: Modeling the neurodynamics of cognition. *Journal of Mathematical Psychology*, *55*(4), 273-289.

Ashby, F. G., & Valentin, V. V. (2007). Computational cognitive neuroscience: Building and testing biologically plausible computational models of neuroscience, neuroimaging, and behavioral data. In M. J. Wenger & C. Schuster (Eds.), *Statistical and process models for cognitive neuroscience and aging* (pp. 15–58). Mahwah, NJ: Erlbaum.

Ashby, F. G., & Waldschmidt, J. G. (2008). Fitting computational models to fMRI data. *Behavior Research Methods*, *40*(3), 713–721.

Bayer, H. M., & Glimcher, P. W. (2005). Midbrain dopamine neurons encode a quantitative reward prediction error signal. *Neuron*, *47*(1), 129-141.

Bayer, H. M., Lau, B., & Glimcher, P. W. (2007). Statistics of midbrain dopamine neuron spike trains in the awake primate. *Journal of Neurophysiology*, *98*(3), 1428–1439.

Bear, M., & Linden, D. (2001). The mechanisms and meaning of long-term synaptic depression in the mammalian brain. In W. Cowan, T. Sudhof, & C. Stevens (Eds.), *Synapses* (pp. 455–517). Baltimore, MD: Johns Hopkins University Press.

Bhalla, U. S., & Bower, J. M. (1993). Exploring parameter space in detailed single neuron models: Simulations of the mitral and granule cells of the olfactory bulb. *Journal of Neurophysiology*, *69*(6), 1948–1965.

Bi, G.-q., & Poo, M.-m. (2001). Synaptic modification by correlated activity: Hebb's postulate revisited. *Annual Review of Neuroscience*, *24*(1), 139–166.

Bogacz, R., Usher, M., Zhang, J., & McClelland, J. L. (2007). Extending a biologically inspired model of choice: Multi-alternatives, nonlinearity and value-based multidimensional choice. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, *362*(1485), 1655–1670.

Bogacz, R., Wagenmakers, E.-J., Forstmann, B. U., & Nieuwenhuis, S. (2010). The neural basis of the speed–accuracy tradeoff. *Trends in Neurosciences*, *33*(1), 10–16.

Boynton, G. M., Engel, S. A., Glover, G. H., & Heeger, D. J. (1996). Linear systems analysis of functional magnetic resonance imaging in human v1. *The Journal of Neuroscience*, *16*(13), 4207–4221.

Cantwell, G., Crossley, M. J., & Ashby, F. G. (2015). Multiple stages of learning in perceptual categorization: Evidence and neurocomputational theory. *Psychonomic Bulletin & Review*, *22*(6), 1598–1613.

Chen, C.-T. (1970). *Introduction to linear system theory*. New York: Holt, Rinehart and Winston.

Clerc, M. (2012). Standard particle swarm optimisation. *Open access archive HAL*.

Cohen, J. D., Braver, T. S., & O'Reilly, R. C. (1996). A computational approach to prefrontal cortex, cognitive control and schizophrenia: Recent developments and current challenges. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, *351*(1346), 1515–1527.

Cohen, J. D., & Servan-Schreiber, D. (1992). Context, cortex, and dopamine: A connectionist approach to behavior and biology in schizophrenia. *Psychological Review*, *99*(1), 45–77.

Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. Cambridge, MA: MIT Press.

Doya, K. (2000). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current Opinion in Neurobiology*, *10*(6), 732–739.

Doya, K. (2007). Reinforcement learning: Computational theory and biological mechanisms. *HFSP Journal*, *1*, 30–40.

Dyson, F. (2004). A meeting with Enrico Fermi. *Nature*, *427*(6972), 297–297.

Ermentrout, G. B. (1996). Type I membranes, phase resetting curves, and synchrony. *Neural Computation*, *8*(5), 979–1001.

Ermentrout, G. B., & Terman, D. H. (2010). *Mathematical foundations of neuroscience*. New York: Springer Science & Business Media.

Estes, W. K. (1950). Toward a statistical theory of learning. *Psychological Review*, *57*(2), 94–107.

Feenstra, M. G., & Botterblom, M. H. (1996). Rapid sampling of extracellular dopamine in the rat prefrontal cortex during food consumption, handling and exposure to novelty. *Brain Research*, *742*(1), 17–24.

Feldman, D. E. (2009). Synaptic mechanisms for plasticity in neocortex. *Annual Review of Neuroscience*, *32*, 33–55.

FitzHugh, R. (1961). Impulses and physiological states in theoretical models of nerve membrane. *Biophysical Journal*, *1*(6), 445–466.

Frank, M. J. (2005). Dynamic dopamine modulation in the basal ganglia: A neurocomputational account of cognitive deficits in medicated and nonmedicated parkinsonism. *Journal of Cognitive Neuroscience*, *17*(1), 51–72.

Frank, M. J., Loughry, B., & O'Reilly, R. C. (2001). Interactions between frontal cortex and basal ganglia in working memory: A computational model. *Cognitive, Affective, & Behavioral Neuroscience*, *1*(2), 137–160.

Grafton, S. T., Hazeltine, E., & Ivry, R. B. (1995). Functional mapping of sequence learning in normal humans. *Journal of Cognitive Neuroscience*, *7*(4), 497–510.

Gu, Q. (2003). Contribution of acetylcholine to visual cortex plasticity. *Neurobiology of Learning and Memory*, *80*(3), 291–301.

Hartley, T., Burgess, N., Lever, C., Cacucci, F., & O'Keefe, J. (2000). Modeling place fields in terms of the cortical inputs to the hippocampus. *Hippocampus*, *10*(4), 369–379.

Haupt, R. L., & Haupt, S. E. (2004). *Practical genetic algorithms*. New York: John Wiley & Sons.

Haykin, S. (2009). *Neural networks and learning machines (3rd ed.)*. Upper Saddle River, NJ: Prentice Hall.

Hélie, S., Paul, E. J., & Ashby, F. G. (2012a). A neurocomputational account of cognitive deficits in Parkinson's disease. *Neuropsychologia*, *50*(9), 2290–2302.

Hélie, S., Paul, E. J., & Ashby, F. G. (2012b). Simulating the effects of dopamine imbalance on cognition: From positive affect to Parkinson's disease. *Neural Networks*, *32*, 74–85.

Helie, S., Roeder, J. L., Vucovich, L., Rünger, D., & Ashby, F. G. (2015). A neurocomputational model of automatic sequence production. *Journal of Cognitive Neuroscience*, *27*(7), 1456-1469.

Hollerman, J. R., & Schultz, W. (1998). Dopamine neurons report an error in the temporal prediction of reward during learning. *Nature Neuroscience*, *1*(4), 304–309.

Houk, J., Adams, J., & Barto, A. (1995). A model of how the basal ganglia generate and use neural signalsthat predict reinforcement. In J. L. Houk J. C. Davis & D. G. Beiser (Eds.), *Models of information processing in the basal ganglia* (pp. 249–270). Cambridge, MA: MIT Press.

Huang, Y.-Z., Edwards, M. J., Rounis, E., Bhatia, K. P., & Rothwell, J. C. (2005). Theta burst stimulation of the human motor cortex. *Neuron*, *45*(2), 201–206.

Huang, Y.-Z., Rothwell, J. C., Chen, R.-S., Lu, C.-S., & Chuang, W.-L. (2011). The theoretical model of theta burst form of repetitive transcranial magnetic stimulation. *Clinical Neurophysiology*, *122*(5), 1011–1018.

Hull, C. (1943). *Principles of behavior*. New York: Appleton-Century-Crofts.

Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, *14*(6), 1569-1572.

Izhikevich, E. M. (2004). Which model to use for cortical spiking neurons? *IEEE Transactions on Neural Networks*, *15*(5), 1063–1070.

Izhikevich, E. M. (2007). *Dynamical systems in neuroscience*. Cambridge, MA: MIT Press.

Kemp, N., & Bashir, Z. I. (2001). Long-term depression: A cascade of induction and expression mechanisms. *Progress in Neurobiology*, *65*(4), 339–365.

Koch, C. (1999). *Biophysics of computation: Information processing in single neurons*. New York: Oxford University Press.

Lapique, L. (1907). Recherches quantitatives sur l'excitation electrique des nerfs traitee comme une polarization. *Journal of Physiology & Pathology: General*, *9*, 620–635.

Lapish, C. C., Kroener, S., Durstewitz, D., Lavin, A., & Seamans, J. K. (2007). The ability of the mesocortical dopamine system to operate in distinct temporal modes. *Psychopharmacology*, *191*(3), 609–625.

Latham, P. E., Richmond, B., Nelson, P., & Nirenberg, S. (2000). Intrinsic dynamics in neuronal networks. I. Theory. *Journal of Neurophysiology*, *83*(2), 808–827.

Lisman, J., Schulman, H., & Cline, H. (2002). The molecular basis of CaMKII function in synaptic and behavioural memory. *Nature Reviews Neuroscience*, *3*(3), 175–190.

Logothetis, N. K. (2003). The underpinnings of the BOLD functional magnetic resonance imaging signal. *The Journal of Neuroscience*, *23*(10), 3963–3971.

Logothetis, N. K., Pauls, J., Augath, M., Trinath, T., & Oeltermann, A. (2001). Neurophysiological investigation of the basis of the fMRI signal. *Nature*, *412*(6843), 150–157.

Maddox, W. T., Bohil, C. J., & Ing, A. D. (2004). Evidence for a procedural-learning-based system in perceptual category learning. *Psychonomic Bulletin & Review*, *11*(5), 945-952.

Maddox, W. T., Glass, B. D., O'Brien, J. B., Filoteo, J. V., & Ashby, F. G. (2010). Category label and response location shifts in category learning. *Psychological Research*, *74*(2), 219-236.

Malenka, R. C., & Siegelbaum, S. A. (2001). Synaptic plasticity. In W. M. Cowan, T. C. Sudhof, & C. F. Stevens (Eds.), *Synapses* (pp. 393–453). Baltimore, MD: Johns Hopkins University Press.

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. New York: Freeman.

Martin, S., Grimwood, P., & Morris, R. (2000). Synaptic plasticity and memory: An evaluation of the hypothesis. *Annual Review of Neuroscience*, *23*(1), 649–711.

McClelland, J. L., McNaughton, B. L., & O'Reilly, R. C. (1995). Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, *102*(3), 419–457.

McCoy, P. A., Huang, H.-S., & Philpot, B. D. (2009). Advances in understanding visual cortex plasticity. *Current Opinion in Neurobiology*, *19*(3), 298–304.

McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, *5*(4), 115–133.

McMillen, T., & Holmes, P. (2006). The dynamics of choice among multiple alternatives. *Journal of Mathematical Psychology*, *50*(1), 30–57.

Meeter, M., Jehee, J., & Murre, J. (2007). Neural models that convince: Model hierarchies and other strategies to bridge the gap between behavior and the brain. *Philosophical Psychology*, *20*(6), 749–772.

Mirenowicz, J., & Schultz, W. (1994). Importance of unpredictability for reward responses in primate dopamine neurons. *Journal of Neurophysiology*, *72*(2), 1024–1027.

Nagumo, J., Arimoto, S., & Yoshizawa, S. (1962). An active pulse transmission line simulating nerve axon. *Proceedings of the IRE*, *50*(10), 2061–2070.

Newell, A., Shaw, J. C., & Simon, H. A. (1958). Elements of a theory of human problem solving. *Psychological Review*, *65*(3), 151–166.

O'Doherty, J. P., Dayan, P., Schultz, J., Deichmann, R., Friston, K., & Dolan, R. J. (2004). Dissociable roles of ventral and dorsal striatum in instrumental conditioning. *Science*, *304*(5669), 452–454.

O'Doherty, J. P., Hampton, A., & Kim, H. (2007). Model-based fMRI and its application to reward learning and decision making. *Annals of the New York Academy of sciences*, *1104*(1), 35–53.

Ogawa, S., Lee, T.-M., Kay, A. R., & Tank, D. W. (1990). Brain magnetic resonance imaging with contrast dependent on blood oxygenation. *Proceedings of the National Academy of Sciences*, *87*(24), 9868–9872.

Øksendal, B. (2003). *Stochastic differential equations*. New York: Springer.

O'Reilly, R. C. (1998). Six principles for biologically based computational models of cortical cognition. *Trends in Cognitive Sciences*, *2*(11), 455–462.

O'Reilly, R. C., Munakata, Y., Frank, M., Hazy, T., et al. (2012). *Computational cognitive neuroscience*. Mainz, Germany: PediaPress.

Paul, E. J., & Ashby, F. G. (2013). A neurocomputational theory of how explicit learning bootstraps early procedural learning. *Frontiers in Computational Neuroscience*, *7*, Article 177.

Pitt, M. A., Kim, W., Navarro, D. J., & Myung, J. I. (2006). Global model analysis by parameter space partitioning. *Psychological*

Review, *113*(1), 57–83.

Pitt, M. A., Myung, I. J., & Zhang, S. (2002). Toward a method of selecting among computational models of cognition. *Psychological Review*, *109*(3), 472–491.

Rall, W. (1967). Distinguishing theoretical synaptic potentials computed for different soma-dendritic distributions of synaptic input. *Journal of Neurophysiology*, *30*(5), 1138-1168.

Rangel, A., & Hare, T. (2010). Neural computations associated with goal-directed choice. *Current Opinion in Neurobiology*, *20*(2), 262–270.

Ricciardi, L. M., & Sacerdote, L. (1979). The Ornstein-Uhlenbeck process as a model for neuronal activity. *Biological Cybernetics*, *35*(1), 1–9.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408.

Rumelhart, D. E., & McClelland, J. L. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1.* Cambridge, MA: MIT Press.

Sandrini, M., Umiltà, C., & Rusconi, E. (2011). The use of transcranial magnetic stimulation in cognitive neuroscience: A new synthesis of methodological issues. *Neuroscience & Biobehavioral Reviews*, *35*(3), 516–536.

Schultz, W. (1998). Predictive reward signal of dopamine neurons. *Journal of Neurophysiology*, *80*(1), 1–27.

Schultz, W. (2002). Getting formal with dopamine and reward. *Neuron*, *26*, 241-263.

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593–1599.

Seamans, J. K., & Robbins, T. W. (2010). Dopamine modulation of the prefrontal cortex and cognitive function. In K. A. Neve (Ed.), *The dopamine receptors, 2nd edition* (pp. 373–398). New York: Springer.

Segev, I., Fleshman, J. W., & Burke, R. E. (1989). Compartmental models of complex neurons. In C. Koch & I. Segev (Eds.), *Methods in neuronal modeling* (pp. 63–96). Cambridge, MA: MIT Press.

Shadlen, M. N., & Newsome, W. T. (2001). Neural basis of a perceptual decision in the parietal cortex (area LIP) of the rhesus monkey. *Journal of Neurophysiology*, *86*(4), 1916–1936.

Sjöström, P. J., Rancz, E. A., Roth, A., & Häusser, M. (2008). Dendritic excitability and synaptic plasticity. *Physiological Reviews*, *88*(2), 769–840.

Smith, P. L., & Ratcliff, R. (2004). Psychology and neurobiology of simple decisions. *Trends in Neurosciences*, *27*(3), 161–168.

Stanton, P. K., Bramham, C., & Scharfman, H. E. (2006). *Synaptic plasticity and transsynaptic signaling*. New York: Springer Science & Business Media.

Strogatz, S. H. (2014). *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering*. Reading, MA: Addison-Wesley.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

Tobler, P. N., Dickinson, A., & Schultz, W. (2003). Coding of predicted reward omission by dopamine neurons in a conditioned inhibition paradigm. *The Journal of Neuroscience*, *23*(32), 10402–10410.

Townsend, J. T., & Ashby, F. G. (1983). *Stochastic modeling of elementary psychological processes*. New York: Cambridge University Press.

Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, *108*(3), 550–592.

Valentin, V. V., Maddox, W. T., & Ashby, F. G. (2016). Dopamine dependence in aggregate feedback learning: A computational cognitive neuroscience approach. *Brain & Cognition*, *109*, 1–18.

Van Zandt, T., Colonius, H., & Proctor, R. W. (2000). A comparison of two response time models applied to perceptual matching. *Psychonomic Bulletin & Review*, *7*(2), 208–256.

Verwey, W. B., Lammens, R., & van Honk, J. (2002). On the role of the SMA in the discrete sequence production task: A TMS study. *Neuropsychologia*, *40*(8), 1268–1276.

Wang, X.-J. (2008). Decision making in recurrent neuronal circuits. *Neuron*, *60*(2), 215–234.

Wiggins, S. (2003). *Introduction to applied nonlinear dynamical systems and chaos* (Vol. 2). New York: Springer Science & Business Media.

Willingham, D. B., Wells, L. A., Farrell, J. M., & Stemwedel, M. E. (2000). Implicit motor sequence learning is represented in response locations. *Memory & Cognition*, *28*(3), 366–375.

Wilson, H. R., & Cowan, J. D. (1972). Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal*, *12*(1), 1–24.

Wilson, H. R., & Cowan, J. D. (1973). A mathematical theory of the functional dynamics of nervous tissue. *Kybernetik*, *13*(2), 55–80.

Yagishita, S., Hayashi-Takagi, A., Ellis-Davies, G. C., Urakubo, H., Ishii, S., & Kasai, H. (2014). A critical time window for dopamine actions on the structural plasticity of dendritic spines. *Science*, *345*(6204), 1616–1620.

Zhang, L. I., Tao, H. W., Holt, C. E., Harris, W. A., & Poo, M.-M. (1998). A critical window for cooperation and competition among developing retinotectal synapses. *Nature*, *395*(6697), 37–44.