How to Build a Brain: A neural architecture for biological cognition

Chris Eliasmith

October 8, 2010

Contents

0.1 Writing tips

- Make it personal: talk about the people in the story, their ideas and quirks

  (esp for history part)

- Make a case: let the reader be the judge, don't 'tell them how it is'.
- Take time to make a point: Writers like gladwell a simple argument that is illustrated with anecdotes; anecdotes must be scientific research
- Use examples: Give an example and then generalize; but the generalization is important
- No math: Wrong audience, provide references, put in appendix
- Pick an audience: Keep the audience consistent for this book: Steve (brother)?; Paul T; Trevor Bekolay; Ronan Reilly
- Citations and notes: Keep them as out of the way as possible, as few notes as possible, all end notes, including citations.
- Be sure to have specific, detailed connection to neural data – distinguish this work from connectionism by biological realism.
- Size: 200-ish pages at 350 wds/pg = 70 000 wds.

0.2 Preface

- use chunks of proposal?
- Book is a bit of a throw back to the good ol days when people used to propose 'unified theories of cognition', or took themselves to do some kind of whole-brain modelling. We have long since been lost in the myriad, amazing, and important details of how specific task are accomplished, and characterizing the mechanisms of relatively small parts of the brain. I think we're in a position to get back to those days. We are no doubt wrong, but good science establishes a fruitful research program... blah?

- really emphasize the teamwork aspect: ‘dream team’ of students & post-docs.
- joke: You may be thinking ’sure, give just about any couple a bit of privancy and about nine months, and they’ll build a brain’. Certainly! And they’ll build a much better one than I can. But, of course I’m not going to tell you how to build a brain the way nature does. I’m going to talk about this in a way that emphasizes a mechanistic understanding of the resulting system. We want to know how to intervene, why certain thigns happen, etc.
- architect metaphor: Though my name is on the cover of this book, taking credit for all of the contents is like an architect taking credit for building a building. I have had a hand in the design, but much of the important work has been done by others. Especially those in my lab... etc.
- this book is a *beginning*, not an end of a research program! (said in intro to chp 3 also)
- Admittedly a grand ambition, blah. i’m an engineer and a philosopher, put them together and you get theo neuro
- remember how the point of cog sci was to understand the whole system... the reason most researchers get into these areas is to understand the brain... but everything is highly focussed, partly understandable, but don’t want to mis the ’big picture’... ala Newell.
- Introduce ‘classical’, and uncertainty about what a cognitive system is:
- The goal of the first chapter is to identify the criteria relevant for distinguish cognitive from non-cognitive systems, and situate them historically.
- some impressive numbers about nengo, number of neurons running on desktop, biggest models run, GPUs and parallel processing, etc.?
- rapid compare/contrast with other simulators? (see reviews)
- mention that the nef is a zeroth order theory in the previous book... this is similarly a start?
- have a section on ‘all the math you need to read this book’: 1. discussion of scalars and vectors, as ’arrows’ or just as points x,y on a graph, as collections of numbers; 2. distinguishing linear from nonlinear functions and

functions in general; 3. integration 4. The math that goes beyond anything described here is in the appendices 5. anything on dynamics? super simple differential equation 6. dot product? 7. any other notation?

As we will see in chapter two, there have been several 'non-classical' attempts at satisfying these criteria. Whether or not these suggestions will ultimately supplant classical architectures, or whether they will merely provide domain specific extensions to such architectures, remains to be seen. What is clear, however, is that the field remains wide open for suggestions as to how we should understand the basic architecture of cognition. The main purpose of this book is to introduce one such possible contender. I call it the "semantic pointer" architecture. A detailed description of this architecture will have to remain until later in the book, after the appropriate groundwork is delayed. Nevertheless, it is useful to keep some of the main motivations for positing this particular view in mind. The first motivation is that most work in cognitive systems seems to be somewhat disconnected from the enormous amount of data that we have about the biological underpinnings of real, living cognitive systems. Now, there has been a realization in some quarters, for instance in many of the projects funded by the EU, that better understanding real biological systems will help us build better cognitive systems. Nevertheless, there are no broadly accepted, systemmatic methods for relating biological data to our high-level understanding of the complex dynamics, and sophisticated representational properties of cognitive systems. In chapter 3, I will describe one systematic, quantified method for realizing exactly this goal.

A second motivation for positing the semantic pointer architecture comes from the realization that once we take seriously the specific computational and representational properties of neural systems, we must adapt our understanding of higher-order cognition to be consistent with those properties. In other words, the architecture I will propose will adopt cognitively relevant representations, computations, and dynamics that are natural to implement in large-scale, biologically realistic neural networks. In short, the semantic pointer architecture is centrally inspired by understanding cognition has a biological process.

Perhaps unsurprisingly, before I can clearly describe this architecture, it will be essential to more completely characterize the other alternatives for understanding cognitive systems. I provide this characterization in two parts: in the remainder of this chapter I will briefly describe the recent history of cognitive science; in the next chapter I will more specifically characterize several of the more influential alternatives currently available for understanding cognitive systems. Throughout this discussion, I'll be attempting to clearly characterize what we mean by a cog-

nitive system, and identify agreed-upon properties of such systems. In chapter 3, I will outline the promised systematic method that allows us to move smoothly from single cell to abstract dynamical descriptions of a neural mechanism, which is called the Neural Engineering Framework (NEF). In chapter 4, I will adopt this framework in order to demonstrate how we can characterize biologically relevant computations and biologically relevant representations for implementing a neurally inspired cognitive architecture. In chapter 5, I describe the semantic pointer architecture in detail. In the remaining two chapters I evaluate the proposed architecture by comparing it to past alternatives. And discuss how research in the science of cognitive systems should proceed.

Chapter 1

The science of cognition

Questions are the house of answers. – Alex, age 5

1.1 The last 50 years

"What have we actually accomplished in the last 50 years of cognitive systems research?" was the pointed question put to a gathering of experts from around the world. They were in Brussels, Belgium, at the headquarters of the European Union funding agency, and were in the process of deciding how to divvy up about 70 million euros of funding. The room went silent. Perhaps the question was unclear. Perhaps so much had been accomplished that it was difficult to know where to start the answer. Or, perhaps even a large room full of experts in the field did not really know any generally acceptable answers to that question.

The point of this particular call for grant applications was to bring together large teams of researchers from disciplines as diverse as neuroscience, computer science, cognitive science, psychology, mathematics, and robotics, in order to unravel the mysteries of how biological cognitive systems are so impressively robust, flexible, adaptive, and intelligent. This was not the first time such a call had been made. Indeed, over the course of the last four or five years this agency has funded a large number of such projects, spending close to half a billion euros. Scientif-

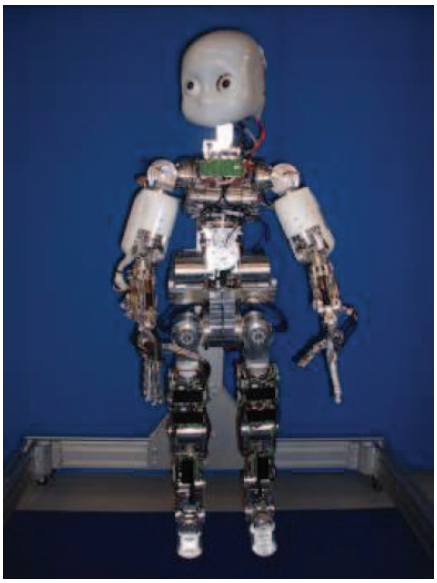

Figure 1.1: The iCub. The iCub is an example of a significant recent advance in robotics. See ???youtube ref??? for videos. ???better picture, permission???

ically speaking, important discoveries have been made by the funded researchers (ff: examples and refs???). However, these discoveries tend not to be of the kind that tell us how to better construct integrated, truly *cognitive* systems. Rather, they are discoveries in the specific disciplines that are taking part in these "Integrated Projects."

For instance, there have been sophisticated new robotic platforms that have been constructed. One example is the iCub (figure 1.1), which is approximately the size of a two-year-old child, and has over 56 degrees of freedom.[1] The iCub has been broadly adopted by researchers in motor control, emotion recognition and synthesis, and active perception (refs???). But, of course, iCub is not a cognitive system. It may be a useful testbed for a cognitive system, no doubt. It may be a wonder of high-tech robotics engineering, indeed. But, it is not a cognitive system.

So the pointed question still stands: "What have we actually accomplished in the last 50 years in cognitive systems research?" That is, what do we now know about how cognitive systems work, that we did not know 50 years ago? Pes-

[1] A degree of freedom is an independent motion of an object. So, moving the wrist up and down is one degree of freedom, as is rotating the wrist, or moving a finger up and down.

simistically, it might be argued that we do not know too much more than we knew 50 years ago. After all, by 1963 Newell and Simon had described in detail the program GPS (General Problem Solver). This program, which was an extension of work that they have started in 1959, is the first in a line of explanations of human cognitive performance that relied on production systems. Production systems are, historically, by far the most influential approach to building cognitive systems. Simply put, a production system consists of a series of productions, or if-then rules, and a control structure. The job of the control structure is to match a given input to 'if' part of these productions to determine an appropriate course of action, captured by the 'then' part.

In 1963, GPS had all of these features. And, it put them to good use. This "program that simulates human thought" (???ref pg number) was able to solve elementary problems in symbolic logic entirely on its own, and went through steps that often matched those reported by people solving the same problem. GPS could be given a novel problem, analyze it, attempt a solution, retrace its steps if it found a dead end (i.e., self-correct), and eventually provide an appropriate solution (Newell and Simon, 1976). (???1963 ref???)

The success of GPS lead to several other cognitive architectures, all of which had production systems as their core. Best known amongst these are Soar(Newell, 1990) and ACT (Adaptive Character of Thought; Anderson, 1983). Despite the many additional extensions to the GPS architecture that were made in these systems, they share a reliance on a production system architecture. The dominance of the representational, computational, and architectural assumptions of production systems has resulted in such systems being called 'classical' approaches to cognitive systems.

While classical approaches had many early successes, such as finding convincing solutions to cryptarithmetic problems and solving complex planning tasks, the underlying architecture does not seem well-suited to interacting with a dynamic, real-world environment, or explaining the evolution of real cognitive systems through time (Eliasmith, 1995). In fairness, more recent work on ACT and ACT-R (Anderson, 1996), has taken dynamics more seriously, explaining reaction times across a wide variety of psychological tasks (Anderson et al., 2004). Nevertheless, explaining reaction times addresses only a very small part of cognitive system dynamics in general, and the ACT explanations rely on a 50ms "cycle time," which itself is not explained.

In fact, those centrally concerned with the dynamics of cognition largely eschew production systems (Port and van Gelder, 1995)(add ref schoner???). If you want to build a real-world cognitive system – one that actually interacts with the

physics of the world – then the most unavoidable constraints on your system are dynamic interactions with the world through perception and action. Roboticists, as a result, seldom use production systems to control their systems. Instead, they carefully characterize the dynamics of their system, attempt to understand how to control such a system when it interacts with the difficult-to-predict dynamics of the world, and look to perception to provide guidance for that control. If-then rules are seldom used. Differential equations, statistics, and signal processing are the methods of choice. Unfortunately, it is unclear how to use those same methods for characterizing high-level *cognitive* behavior, like language, complex planning, and deductive reasoning – behaviors for which traditional approaches have the most success.

In short, there is a broad gap in our understanding of real, cognitive systems: on the one hand there are the approaches taken to understanding dynamic, real-world perception and action; on the other hand there are the approaches taken to understanding higher-level cognition. Unfortunately, these approaches are not the same. Nevertheless, it is obvious that perception/action and high-level cognition are not two independent parts of one system. Instead, these two aspects are, in some fundamental way, integrated in cognitive animals such as ourselves.[2] Indeed, a major theme of this book is to suggest that it is through biology that we will be able to understand this integration. But for now, I am only concerned to point out that classical architectures are not obviously appropriate for understanding all aspects of real cognitive systems. This, then, is why we cannot simply say, in answer to the question of what has been accomplished in the last 50 years, that we have identified *the* (classical) cognitive architecture.

However, this does not mean that identifying such architectures is without merit. On the contrary, one undeniably fruitful consequence of the volumes of work that surrounds the discussion of classical architectures is the identification of criteria for what counts as a cognitive system. That is, when proponents of classicism were derided for ignoring cognitive dynamics, one of their most powerful responses was to note that their critics had no truly *cognitive* systems to replace theirs with. This resulted in a much clearer specification of what a cognitive system was. So, I suspect that there would be agreement amongst most of the experts gathered in Brussels as to what has been accomplished lo these 50 years. Indeed, the accomplishments are not in the expected form of an obvious technology, a breakthrough method, or an agreed upon theory. Instead, the major accomplishments have been in clarifying what the questions are, in determining what counts

[2]For standard discussions at this point see (refs???).

as a cognitive system, and in figuring out how we are most likely to succeed in explaining such systems (or, perhaps more accurately, how we are *not* likely to succeed).

If true, this is no mean feat. Indeed, it is even more true in science than elsewhere that, as economic Nobel laureate Paul A. Samuelson has observed, "good questions outrank easy answers." If we actually have more thoroughly identified criteria for distinguishing cognitive from non-cognitive systems, and if we really have a good sense of what methods will allow us to understand how and why systems can successfully meet those criteria, we have accomplished a lot. Ultimately, only time will tell if we are on the right track. Nevertheless, I believe there is an overall sense in the field that we have a better idea of what we are looking for in an explanation of a cognitive system than we did 50 years ago – even if we do not yet know what that explanation is. Often, progress in science is more about identifying specific questions that have uncertain answers than it is about proposing specific answers to uncertain questions.

The goal of this first chapter, then, is to identify these cognitive criteria and articulate some questions arising from them. These appear in sections 1.3 and 1.4 respectively. First, however, it is worth a brief side trip into the history of the cognitive sciences to situate the concepts and methods that have given rise to these criteria.

1.2 How we got here

In the previous section, I identified the 'classical' approach to understanding cognition. And, I contrasted this approach with one that is more centrally interested in the characterization of the dynamics of behavior. However, much more needs to be said about the relationship between classical and non-classical approaches in order to get a general lay-of-the-land in cognitive systems theorizing. Indeed, much more can be said than I will say here (see, e.g., Bechtel and Graham, 1999). My intent is to introduce the main approaches in order: 1) to identify the strengths and weaknesses of these approaches, both individually and collectively; 1) to state and clarify the cognitive criteria mentioned earlier; and, ultimately, 3) to outline a novel theory of biological cognition in later chapters.

In the last half century, there have been three major approaches to theoriz-

ing about the nature of cognition. Each approach has relied heavily on a preferred metaphor for understanding the mind/brain. Most famously, the classical approach (aka 'symbolicism', or Good Old-fashioned Artificial Intelligence (GOFAI)), relies on the "mind as computer" metaphor. Under this view, the mind is the software of the brain. Jerry Fodor, for one, has argued that the impressive theoretical power provided by this metaphor is good reason to suppose that cognitive systems have a symbolic "language of thought" which, like a computer programming language, expresses the rules that the system follows (Fodor, 1975). Fodor claims that this metaphor is essential for providing a useful account of how the mind works. Production systems, which I have already introduced, have become the preferred implementation of this metaphor.

A second major approach is often called 'connectionism' (aka the Parallel Distributed Processing (PDP) approach or New-fangled Artificial Intelligence (NFAI)). In short, connectionists explain cognitive phenomena by identifying large networks of typically identical nodes, that are connected together in various patterns. Each node performs a simple input/output mapping. However, when grouped together in sufficiently large networks, the activity of these nodes is interpreted as implementing rules, analyzing patterns, or performing any of several other cognitively-relevant behaviors. Connectionists, like the symbolicists, rely on a metaphor for providing explanations of cognitive behaviors. This metaphor, however, is much more subtle than the symbolicist one; these researchers presume that the functioning of the mind is like the functioning of the brain. The subtlety of the "mind *as* brain" metaphor lies in the fact that connectionists also hold that the mind *is* the brain. However, when providing *cognitive* descriptions, it is the metaphor that matters, not the identity. In deference to the metaphor, the founders of this approach call it "brain-style" processing, and claim to be discussing "abstract networks" (Rumelhart and McClelland, 1986). In other words, their models are not supposed to be direct implementations of neural processing, and hence cannot be directly compared to the kinds of data we gather from real brains. This is not surprising since the computational and representational properties of the nodes in connectionist networks bear little resemblance to neurons in real biological neural networks.[3]

The final major approach to cognitive systems in contemporary cognitive science can be called 'dynamicism,' and is often identified with 'embedded' or 'embodied' approaches to cognition. Proponents of dynamicism also rely heavily on a metaphor for understanding cognitive systems. Most explicitly, van Gelder

[3] As discussed in chapter 10 of (Bechtel and Abrahamsen, 2001).

employs the Watt Governor as a metaphor for how we should characterize the mind (van Gelder, 1995). It is through his analysis of the best way to characterize this dynamic system that van Gelder argues for understanding cognitive systems as non-representational, low-dimensional, dynamic systems. Like the Watt Governor, van Gelder maintains, cognitive systems are essentially dynamic and can only be properly understood by characterizing their state changes through time. The "mind as Watt Governor" metaphor suggests that trying to impose any kind of discreteness, either temporal or representational, will lead to a mischaracterization of cognitive systems. This same sort of analysis – one which highlights the importance of dynamics – highlights the essential coupling of cognitive systems to their environment (van Gelder and Port, 1995). Dynamic constraints are clearly imposed by the environment on the success of our behavior (we must see and avoid the cheetah before it eats us). If our high-level behaviors are built on our low-level competencies, then it is not surprising that identifying this important role of the environment has lead several researchers to emphasize the fact that real cognitive systems are embedded within a specific environment, with specific dynamics. Furthermore, they have argued, the nature of that environment can have significant impacts on what cognitive behaviors are realized (ref andy??? others???). Because many of the methods and assumptions of dynamicsm and embedded approaches are shared, in this discussion I group both under the heading of 'dynamicism.'

Notably, each of symbolicism, connectionism, and dynamicism, rely on metaphor not only for explanatory purposes, but also for developing the conceptual foundations of their preferred approach to cognitive systems. For symbolicists, the properties of Turing machines become shared with minds because both are computational systems. For connectionists, the character of representation changes dramatically under their perferred metaphor. Mental representations are taken to consist of "sub-symbols" associated with each node, while "whole" representations are real-valued vectors in a high-dimensional property space.[4] Finally, because the Watt Governor is best described by the mathematics of dynamic systems theory, which makes no reference to computation or representation, dynamicists claim that our theories of mind need not appeal to computation or representation either (van Gelder, 1998).

I have argued elsewhere that our understanding of cognitive systems needs

[4]See, for example, (Smolensky, 1988). Notably, there are also connectionist models that take activities of individual nodes to be representations. These are still very much unlike symbolic representations.

to move beyond such metaphors (Eliasmith, 2003). We need to move beyond metaphros because, in science, metaphorical thinking can sometimes unduly constrain available hypotheses. This is not to deny that metaphors are incredibly useful tools at many points during the development of scientific theory. It is only to say that, sometimes, metaphors only go so far. Take, for instance, the development of the current theory of the nature of light. In the nineteenth century, light was understood in terms of two metaphors: light as a wave, and light as a particle. Thomas Young was the best known proponent of the first view, and Isaac Newton was the best known proponent of the second. Each used their favored metaphor to suggest new experiments, and develop new predictions.[5] Thus, these metaphors played a role similar to that played by the metaphors discussed above in contemporary cognitive science. However, as we know in the case of light, both metaphors are false. Hence the famed "wave-particle duality" of light: sometimes it behaves like a particle; and sometimes it behaves like a wave. Neither metaphor by itself captures all the phenomena displayed by light, but both are extremely useful in characterizing some of those phenomena. So, understanding what light *is* required moving beyond the metaphors.

I believe that the same is true in the case of cognition. Each of the metaphors mentioned above has some insights to offer regarding certain phenomena displayed by cognitive systems. However, none of these metaphors is likely lead us to all of the right answers. Thus, we ideally want a way of understanding cognitive systems that draws on the strengths of symbolicism, connectionism, and dynamicism, but does not depend on the metaphors underlying these approaches.

In fact, I believe we are in a historically unique position to affect this kind of understanding. This is because we currently have more detailed access to the underlying biology of cognition than ever before. We can record large-scale blood flow in the brain during behavior using fMRI, we can image medium-sized networks of cells during the awake behavior of animals, we can record many individual neurons inside and outside of the functioning brain, and we can even record the opening and closing of individual channels, which are about 200 nanometers in size, on the surface of a single neural cell.

This kind of information is exactly what we need to free ourselves of high-level guiding metaphors. To be clear, the preceding sentence is not the claim that we should get rid of all metaphors. I by no means think that is plausible. Instead, I am advocating that we reduce our reliance on metphors *as theoretical arbiters*.

[5] For a detailed description of the analogies, predictions, and experiments, see (Eliasmith and Thagard, 1997).

That is, we need to remove them from a role in determining what is a good explanation and what is not. If we can relate our cognitive explantions to specific biological measurements, then it is that data, not a metaphor, that should be the determiner of the goodness of our explanation. If we can identify the relationship between cognitive theories and biological data, then we can begin to understand cognitive systems for what they are: complex, dynamic, biological systems. The three past approaches I have discussed provide few, if any, hints as to the nature of this relationship.

One reason such hints are scarce is that, historically speaking, theories regarding cognitive systems mainly come from psychology, an offshoot of philosophy of mind in the late 19^{th} century. At the time, very little was known of the biology of the nervous system. There was some suggestion that the brain was composed of neurons, but this was highly contentious. Certainly there was little understanding of how cells could be organized to control our simple behaviors, let alone our complex, cognitive ones.

So, cognitive theorizing has proceeded without much thought about biology. In the early days of psychology, most such theories were generated by the method of introspection: i.e., sitting and thinking very carefully, so as to discern the components of cognitive processing. Unfortunately, different people introspected different things, and so there was a crises in 'introspectionist' psychology. This crisis was resolved by 'behaviorist' psychologists who simply disallowed introspection. The only relevant data for understanding cognitive systems was data that could be gleaned from the 'outside', i.e., from behavior.

While much is sometimes made of the difference between 'philosophical' and 'psychological' behaviorism, there was general agreement on this much: internal representations, states, and structures are irrelevant for understanding the behavior of cognitive systems. For psychologists, like Watson and Skinner, this was true because only input/output relations are scientifically accessible. For philosophers, like Ryle, this was true because mental predicates (like 'believes', 'wants', etc.), if they were to be consistent with natural science, must be analyzable in terms of behavioral predicates. In either case, looking inside the "black box" that was the system of study, was prohibited.

Interestingly, engineers of the day respected a similar constraint. In order to understand dynamic physical systems, the central tool they employed was classical control theory. Classical control theory, perhaps notoriously, only characterizes physical systems in terms of their input/output relationship. As a result, classical control theory is limited to designing non-optimal, single-variable, static controllers and depends on graphical methods, rules of thumb, and does not allow

for the inclusion of noise.[6] While the limitations of classical controllers and methods are now well-known, they nevertheless allowed engineers to build systems of kinds they had not systematically built before: goal-directed systems.

While classical control theory was practically useful, especially in the 1940s when there was often a desire to blow things up (the typical goal at which such systems were directed), some researchers thought the theory had more to offer. They suggested that classical control theory could provide a foundation for describing living systems as well. Most famously, the interdisciplinary movement founded in the early 1940s known as 'cybernetics' was based on precisely this contention.[7] Cyberneticists claimed that living systems, like classical control systems, were essentially goal-directed systems. Thus, closed-loop control should be a good way to understand the behavior of living systems. Given the nature of classical control theory, cyberneticists focused on characterizing the input/output behavior of living systems, not their internal processes. Unfortunately for cyberneticists, in the mid-50s there was a massive shift in how cognitive systems were viewed.

With the publication of a series of seminal papers, [8] the 'cognitive revolution' took place. One simplistic way to characterize the move from behaviorism to 'cognitivism' is that it became no longer taboo to look inside the black box. Quite the contrary: internal states, internal processes, and internal representations became standard fare when thinking about the mind. Classical control theory no longer offered the kinds of analytic tools that mapped easily onto this new way of conceiving complex biological behavior. Instead, making sense of the insides of that black box was heavily influenced by concurrent successes in building and programming computers to perform complex tasks. Thus, many early cognitive scientists saw, when they opened the lid of the box, a computer. As explored in detail by Jerry Fodor, "[c]omputers show us how to connect semantical [meaning-related] with causal properties for *symbols*" (Fodor, 1987, p. 18), thus computers have what it takes to be minds. Once cognitive scientists began to think of minds as computers, a number of new theoretical tools became available for characterizing cognition. For instance, the computer's theoretical counterpart, the Turing machine, suggested novel philosophical theses, including 'functionalism' (the notion that only the mathematical function computed by a system was relevant for its being a mind or not) and 'multiple realizability' (the notion that a mind could

[6] For a succinct description of the history of control theory, see (Lewis, 1992).

[7] For a statement of the motivations of cybernetics, see (Rosenblueth et al., 1943).

[8] These papers include, but are not limited to Newell et al. (1958), Miller (1956), Bruner et al. (1956), and Chomsky (1959).

be implemented (i.e. realized), in pretty much any substrate – water, silicon, interstellar gas, you name it – as long as it computed the appropriate function). More practically, the typical architecture of computers, the von Neumann architecture (which is a predecessor of the architecture of a production system), was thought by many to be relevant for understanding our cognitive architecture.

Eventually, however, adoption of the von Neumann architecture for understanding minds was seen by many as poorly motivated. Consequently, the early 1980s saw a significant increase in interest in the connectionist research program. As mentioned previously, rather than adopting the architecture of a digital computer, these researchers felt that an architecture more like that seen in the brain would provided a better model for cognition. It was also demonstrated that a connectionist architecture could be as computationally powerful as any symbolicist architecture (ref???). But despite the similar computational power of the approaches, the specific problems at which each approach excelled were quite different. Connectionists, unlike their symbolicist counterparts, were very successful at building models that could learn and generalize over the statistical structure of their input. Thus, they could begin to explain many phenomena not easily captured by symbolicists, such as object recognition, reading, concept learning, and other behaviors crucial to cognition.

For some, however, connectionists had clearly not escaped the influence of the mind-as-computer metaphor. Connectionists still spoke of representations, and thought of the mind as a kind of computer. These dynamicists, who we have met before, suggested that if we want to know which functions a system can actually perform in the real world, we must know how to characterize the system's dynamics. Consequently, since cognitive systems evolved in dynamic environments, we should expect evolved control systems, like brains, to be more like the Watt Governor – dynamic, continuous, coupled directly to what they control – than like a discrete-state Turing machine that computes over 'disconnected' representations. As a result, dynamicists suggested that dynamic systems theory, not computational theory, was the right quantitative tool for understanding minds. They claimed that notions like 'chaos,' 'hysteresis,' 'attractors,' and 'state-space' underwrite the conceptual tools best-suited for describing cognitive systems.

So, as we have just seen, each of the three positions grew out of critical evaluation of previous positions. Connectionism was a reaction to the over-reliance on computer architectures for describing cognition. Dynamicism was a reaction to an under-reliance on the importance of time and our connection to the physical environment. Even symbolicism was a reaction to its precursor, behaviorism, which had ruled out a characterization of cognition which posited states internal to the

agent. Consequently, each of the metaphors have been chosen to emphasize different aspects of cognition, and hence driven researchers in these areas to employ different formalisms for describing their cognitive theories. Simply put, symbolicists use production systems, connectionists use networks of simple nodes, and dynamicists use sets of differential equations to explain our most complex behaviors.

While we can see the progression through these approaches as rejections of their predecessors, it is also important to note what was preserved. Symbolicism preserved the commitment to providing scientific explanations of cognitive systems. Connectionism retained a commitment to the notions of representation and computation so central to symbolicist approaches. And dynamicism, perhaps the most self-conscious attempt to break away from previous methods, has not truly broken with tradition. Rather, dynamicists have most convincingly argued for a shift in emphasis: they have made time a non-negotiable feature of cognition.[9]

Perhaps, then, a successful break from all of these metaphors will be able to relate each of the central methods to one another in a way that preserves the central insights of each. And, just as importantly, such a break from past approaches should allow us to see how our theories relate to the plethora of data we have about brain function. In my more impetuous moments, I have been known to argue that the methods developed in this book are one such break. I have such moment in section ???. Before such arguments can be made with any force, however, much work needs to be done. First, for instance, I need to use the historical background just presented to identify what I earlier claimed was the major advance of the the last 50 years: identifying criteria for distinguishing cognitive from non-cognitive systems.

1.3 Where we are

It will strike many as strange to suggest that there is anything like agreement on criteria for identifying cognitive systems. After all, this area of research has been dominated by, shall we say, "vigorous debate" between proponents of each of the three approaches. It is true that there are instances of symbolicists calling connectionism "quite dreary and recidivist" (Fodor, 1995). Nevertheless, I believe

[9]However, it should be noted that Newell, one of the main developers of production systems, was quite concerned with the timing of cognitive behavior. He dedicated a large portion of his magnum opus, *Unified Theories of Cognition*, to the topic. Nevertheless, time is not a necessary feature of production systems, and so his considerations seem, in may ways, after the fact (Eliasmith, 1996).

that there is some agreement on what the target of explanation is. By this I do not mean to suggest that there is an agreed-upon definition of cognition. But rather that 'cognition' is to most researchers what 'pornography' was to justice Potter Stewart: "I shall not today attempt further to define [pornography]; and perhaps I could never succeed in intelligibly doing so. *But I know it when I see it*."

Researchers seem to know cognition when they see it, as well. There are many eloquent descriptions of various kinds of behavior in the literature, which most readers – regardless of their commitments to symbolicism, connectionism, or dynamicism – recognize as cognitively relevant behavior. It is not as if symbolicists think that constructing an analogy is a cognitive behavior and dynamicists disagree. This is why I have suggested that we may be able to identify agreed-upon criteria for the characterization of cognitive systems. To be somewhat provocative, I will call these the 'Quintessential Cognitive Criteria,' or 'QCC' for short. I should note that this section is only a first pass and summary of considerations for the proposed QCC in table 1.1. I return to a much more detailed discussion of them before explicitly employing them in chapter 8.

Criteria are, of course, not necessary and sufficient conditions that a system must meet to be cognitive. They are not, in other words, definitive. Instead, they are standards against which a system can be judged to determine to what extent it has properties relevant for cognition. Since criteria are not necessary and sufficient conditions, employing them makes it natural to claim that some systems are 'more cognitive' than others. This suggests that there is a 'cognitive continuum.' A rough ordering of some common research subjects in terms of their 'cognitiveness' might be: humans, chimpanzees, monkeys, cats, rats, turtles, flies, and worms.[10] If we appropriately identify the QCC, they should be consistent with this kind of intuitive ordering. In other words, to place a system on the cognitive continuum, we can appeal to the number and degree to which the QCC are met: the more and the better the QCC are met, the more cognitive the system will be.

So what are the QCC? Let us turn to what have researchers said about what makes a system cognitive. Here are examples from proponents of each view:

- Dynamicism (van Gelder, 1995, p. 375-6): "[C]ognition is distinguished from other kinds of complex natural processes... by at least two deep features: on the one hand, a dependence on knowledge; and distinctive kinds of complexity, as manifested most clearly in the structural complexity of

[10] As a matter of interest, I conducted an informal poll of 20 researchers spanning the cognitive disciplines, and 18 gave this response. One response was incomplete, and the other placed turtles ahead of rats.

natural languages.".

- Connectionism (McClelland and Rumelhart, 1986, p. 13): Rumelhart and McClelland explicitly identify the target of their well-known PDP research as "cognition." To address it, they feel that they must explain "motor control, perception, memory, and language".

- Symbolicism (Newell, 1990, p. 15): Newell presents the following list of behaviors in order of their centrality to cognition: 1) problem solving, decision making, routine action; 2) memory, learning, skill; 3) perception, motor behavior; 4) language; 5) motivation, emotion; 6) imagining, dreaming, daydreaming.

These examples do not provide criteria for identifying cognitive systems, but rather attempt to identify which particular aspects of behavior must be explained in order to successfully explain cognition. However, from such lists we can distill the QCC that capture the intuitions motivating these researchers. Notice that there are several commonalities among the lists. First, language appears in all three. This is no surprise as language is often taken to be the pinnacle of human cognitive ability. But, there are other important shared commitments evident in these lists. For instance, each identify the importance of adaptability and flexibility. For van Gelder adaptability is evident through his identification of the dependence of cognitive behavior on knowledge. For the other two, explicit mention of memory and learning highlight a more general interest in adaptability. As well, while not in this specific quote from van Gelder, dynamicism is built on a commitment to the centrality of action and perception to cognition (Port and van Gelder, 1995). Perhaps surprisingly, then, it is the connectionists and symbolicist who clearly identify motor control and perception as important to understanding cognition. In any case, it is clear that all three agree on the important role of these, more basic, processes.

While these are simple lists, I believe we can see in them the motivations for subsequent discussions explicitly aimed at identifying what it takes for a system to be cognitive. Let us briefly consider some of these more direct discussions. One of the first, and perhaps the most well-known discussions, is provided by Fodor and Pylyshyn in their 1988 paper "Connectionism and cognitive architecture: A critical analysis." While mainly a critique of the connectionism of the day, this paper also provides three explicit constraints on what it takes to be a cognitive system. These are productivity, systematicity, and compositionality. Productivity is the ability of a system to generate a large number of representations based on a

few basic representations (a lexicon) and rules for combining them (a grammar). Systematicity refers to the fact that some sets of representations (generated productively) come together. For instance, they suggest that cognitive systems cannot represent 'John loves Mary' with out thereby being able to represent 'Mary loves John'. Finally, compositionality is the suggestion that the meaning of complex representations is a direct 'composition' (i.e. adding together) of the meanings of the basic representations.

More recently, Jackendoff has dedicated his book to identifying challenges for a cognitive neuroscience of cognition (Jackendoff, 2002). In it he suggests that there are four main challenges to address when explaining cognition. Specifically, Jackendoff's challenges are: 1) the massiveness of the binding problem (that very many basic representations must be bound to construct a complex representation); 2) the problem of 2 (how multiple instances of one representational token can be distinguished); 3) the problem of variables (how can roles (e.g. 'subject') in a complex representation be generically represented); and 4) how to incorporate long-term and working memory into cognition. Some of these challenges are closely related to those of Fodor and Pylyshyn, and so are integrated with them as appropriate in the QCC (see table 1.1).

The Fodor, Pylyshyn, and Jackendoff criteria come from a classical, symbolicist perspective. In a more connectionist-oriented discussion, Don Norman summarizes several papers he wrote with Bobrow in the mid-70s, in which they argue for the essential properties of human information processing (???ref). Based on their consideration of behavioral data, they argue that human cognition is: robust (appropriately insensitive to missing or noisy data, and damage to its parts), flexible, and relies on 'content-addressable' memory. Compared to symbolicist considerations, the emphasis in these criteria has moved from representational constraints to more behavioral constraints, driven by the 'messiness' of psychological data.

Dynamicists can be seen to continue this trend towards complexity in their discussions of cognition. Take, for instance, Gregor Schoner's discussion in his article "Dynamical systems approaches to cognition" (???ref). In his opening paragraphs, he provides examples of the sophisticated action and perception that occurs during painting, and playing in a playground. He concludes that "cognition takes place when organisms with bodies and sensory systems are situated in structured environments, to which they bring their individual behavioral history and to which they quickly adjust" (p. 101). Again, we see the importance of flexibility and robustness, with the addition of an emphasis on the role of the environment.

Before presenting a final summary of the QCC, I believe there are additional

criteria that a good theory of cognition must meet. I suspect that these will not be controversial, as they are a summary of insights that philosophers of science have generated in their considerations of what constitutes a good scientific theory in general (refs???). I take it as too obvious to bother stating that each approach assumes that we are trying to construct a scientific theory cognitive systems. Nevertheless, these criteria may play an important role in distinguishing good cognitive theories from bad.

Two of the most important considerations for good scientific theories are those of unity and simplicity. Good scientific theories are typically taken to be unified: the more sources of data, and the more scientific disciplines that they are consistent with, the better the theory. One of the reasons Einstein's theory of relativity is to be preferred over Newton's theories of motion is that the former is consistent with more of our observations. That is, we can triangulate our data sources in such a way as to prefer one over the other. In addition, good theories tend to simplicity. That is good theories can be stated compactly. The reason that the heliocentric theory of our solar system is to be preferred over a geocentric one is that, in the latter, we need to specify not only the circular paths of the planets, but also the many infamous 'epicycles' of each planet in order to explain their motions. In contrast, the heliocentric theory needs to specify one simple ellipse for each planet.

Though this discussion has been brief, I believe that these considerations, coupled with the earlier evidence of a convergence of an understanding of cognitive systems, provide a reasonably clear indication of several criteria that researchers in each of the three approaches would agree to. As a result, table 1.1 summarizes the QCC we can, at least on the face of it, extract from this discussion. As a reminder, I do not expect the mere identification of these criteria to be convincing. A detailed discussion of each is presented in chapter 8.

1.4 The house of answers

While the QCC should prove useful for evaluating a characterization of cognition, they are not obviously useful for *directing* such a characterization. Instead, I think we need to turn to specifying a few central questions that have arisen in the last 50 years of cognitive science research. As a result, in this section I suggest four questions which, if answered in detail, should go a long way to providing a characterization of a cognitive system that addresses most, if not all, of the QCC.

These questions are:

1. Representational structure
a. Systematicity b. Compositionality c. Productivity (the problem of variables) d. The massive binding problem (the problem of two)
2. Performance concerns
a. Syntactic generalization b. Robustness c. Adaptability d. Memory
3. Scientific merit
a. Triangulation (Contact with more sources of data) b. Compactness

Table 1.1: Quintessential Cognitive Criteria (QCC) for theories of cognition.

1. How are semantics captured by the system?
2. How is syntactic structure encoded and manipulated by the system?
3. How is information flexibly routed through the system in response to task demands?
4. How are memory and learning accounted for by the system?

A long-standing concern when constructing models of cognitive systems is how to characterize the relationship between the representations inside the system, and the objects in the external world which they purportedly represent. That is, how do we know what a representation means? Of course, for any system we construct, we can simply define what the meaning of particular representation is. Unfortunately, this is very difficult to do convincingly: consider trying to define a mapping between dogs-in-the-world and a state in your head that acts like our concept 'dog'. The vast psychological literature on concepts points to the complexity of this kind of mapping. Most researchers in cognitive science are well aware of, and dread, addressing this problem, which is often called the 'symbol grounding problem' (ref?). Nevertheless, any implemented model of a cognitive system must make some assumptions about how the representations in that system get their meaning. Consequently, answering this first question will force anyone wishing to characterize cognition to, at the very least, state their assumptions about how

internal states are related to external ones. Whatever the story, it will have to plausibly apply not only to the models we construct, but also the natural systems we are attempting to explain.

The second question addresses what I have already noted is identified nearly universally as a hallmark of cognitive systems: the manipulation of structured representations. Whether we think *internal* representations themselves are structured (or, for that matter, if they even exist), we must face the undeniable ubiquity of behavior involving the manipulation of language-like representations. Consequently, any characterization of cognition will have to tell a story about how syntactic structure is encoded and manipulated by a cognitive system. Answering this question will unavoidably address at least the first five criteria in the QCC.

Another broadly admired feature of cognitive systems is their incredible, rapid adaptability. People put into a new situation can quickly survey their surroundings, identify problems, and formulate plans for solving those problems – often within the space of a few minutes. Performing each of these steps demands coordinating the flow of huge amounts of information through the system. Even seemingly mundane changes in our environment, such as switching from a pencil to a marker while writing, pose difficult control problems. Such a simple switch can alter the weight of the hand, change what are valid configurations of the fingers, and modify the expected visual, auditory, and proprioceptive feedback signals. All of these pieces of information can influence what is the best motor control plan, and so must be taken into account when constructing such a plan. Because the sources of such information can remain the same (e.g. visual information comes from the visual system), while the destination of the information may change (e.g., from arm control to finger control), a means of routing that information must be in place. More cognitively speaking, if I simply tell you that the most relevant information for performing a task is going to switch from something you are hearing to something you will be seeing, you can instantly reconfigure the information flow through your brain to take advantage of that knowledge. Somehow, you are re-routing the information you use for planning to come from the visual system instead of the auditory system. We do this effortlessly, we do this quickly, and we do this constantly.

The fourth question focuses our attention on another important source of cognitive flexibility: our ability to use past information to improve our performance on a future task. The timescale over which information might be relevant to a task ranges from seconds to many years. Consequently, it is not surprising that the brain has developed mechanisms that also range over these timescales. Memory and learning are behavioral descriptions of the impressive abilities of these

mechanisms. Considerations of memory and learning directly address several of the performance concerns identified in the QCC. Consequently, any characterization of a cognitive system has to provide some explanation for how relevant information is propagated through time, and how the system can adapt to its past experience.

As mentioned earlier, I believe the great success of cognitive science is the improved ability to formulate questions that need to be addressed when studying cognition. The four broad questions I have described here are really intended to highlight much larger classes of more specific, and hence more useful, questions. (And those of you who read the table of contents may notice that they also align with the aims of chapters 3 through 6, respectively.) The purpose of only identifying these four as central to characterizing a cognitive system, is to avoid the possibility of getting lost in the details of a laundry list of detailed questions relevant to cognition. Nevertheless, compiling such a list can be helpful. In table 1.2 I provide a small snippet of such a list.

Having lists of questions like this can prove helpful for guiding our discussion of cognitive systems, but it does not suggest a way we might go about answering them. However, I think we can take an approach in cognitive science that has been espoused as crucial for other sciences. That is, we can give a good characterization of a cognitive system if we show how to build one.

Richard Feynman, the famous physicist, wrote a few of his last, perhaps most dear, thoughts on the blackboard in his office shortly before he died. After his death, someone had the foresight to take a photograph of his blackboard for posterity, which you can see at `http://abbynuss1.tripod.com/id32.htm`. In large letters, in the top left corner, he wrote:

> *What I cannot create I do not understand.*

I would like to propose to adopt this as a motto for understanding cognition. It suggests that *creating* a cognitive system would provide one of the most convincing demonstrations that we truly understand such a system. Of course, in this case, as in the case of many other physical systems, 'creating' the system amounts to creating simulations of the underlying mechanisms in detail. Although I will not argue for this point in great depth – though many others have (refs? dretske?, mech people) – but, hopefully an explanation of a cognitive system that begins with neurons as component parts, characterizes those basic mechanisms quantitatively, and demonstrates how they can be arranged to give rise to a wide variety of cognitive behaviors will prove compelling evidence for the claim that such a characterization provides us with a improved understanding of cognitive systems.

Table 1.2: A partial laundry list of important questions that should be addressed by an account of cognition.

Semantics questions	
	Can the semantic account be the same for all neural representations (i.e., from light intensity up to fu
	Are there amodal representations, how is their content determined?
	What is the role and importance of hierarchies in visual/motor systems to representational content?
	What is the relation between an increase in hierarchical level and an increase in representational com
	Does our semantic characterization (have to) solve the frame problem?
Syntax questions	
	What kind of neurally realistic architectures can support syntactic structure?
	What kinds of evidence can we bring to bear to distinguish approaches to neural binding?
	Does syntactic binding use the same mechanism as perceptual binding? What is the relation between
	Is our chosen syntactic mechanism fast enough to account for on-the-fly structure processing?
	Can our syntactic mechanism scale to allow efficient processing of vocabularies of the size employed
Control questions	
	How does such a seemingly distributed system act in such a unified way?
	How can we efficiently deal with the curse of dimensionality?
	How, in a generic sense, are neurons organized to give rise to dynamics and computations that are use
	Does the architecture decompose complex control problems (if so, how), or have specialized controll
	What is the precise nature of the subtle intercommunication between perceptual, cognitive, and moto
Memory and learning questions	
	How does the wide variety of timescales relevant for cognition arise from a brain, whose individual c
	How are internal signals that drive adaptation generated?
	What kind of architecture can be adaptable in the many ways that brains are? (from learning fine-mot
	What kinds of memory are there, and how do these map onto brain mechanisms?
	How much of our proposed architecture can be learned and how much (and what precisely) must be i
Other questions	
	How do 'low-level' properties of systems affect 'high-level' function? (What do we mean by low and
	How do so few cell types do so many different things?
	How do 'levels' of description relate given the vast range of spatial and temporal scales of available d
	Why do the same brain areas 'light up' during vastly different tasks?
	Does the architecture support the right kinds of interactions with its environment over the right time f

This, in a nutshell, is the goal I'm striving towards. In the next chapter, I begin at the bottom with neural mechanisms, so we can begin to work our way towards an understanding of the neural basis of cognition.

1.5 Nengo: An introduction

This is the first of the 10 tutorials that are included in this book. There is one tutorial at the end of each chapter, and each provides a instructions for constructing a simulation related to a key concept encountered in that chapter. All of the tutorials use the Nengo software package, which is actively supported by my lab. Tutorials can be skipped without interrupting the flow of the rest of the book.

Nengo (Neural ENGineering Objects) is a graphical neural simulation environment developed in my lab. In this section, I describe how to install and use Nengo, and guide you through simulating a single neuron. The simulation you will build is shown in Figure 1.2. Anything you must do is placed on a single bulleted line. Surrounding text describes what you are doing in more detail. All simulations can also simply be loaded through the *File* menu. They are in the 'Building a Brain' subdirectory.

- mention youtube flythrough/demos or other videos?

Installation

Nengo works on Mac, PC, and Linux machines and can be installed by downloading the software from `http://nengo.ca/`.

- To install, unzip the downloaded file where you want to run the application from.
- In the unzipped directory, double-click 'Nengo' to run the program.

An empty Nengo world appears. This is the main interface for graphical model construction.

Building a model

The first model we will build is very simple: just a single neuron.

- Right-click anywhere on the background and choose *New Network.* Set the *Name* of the network to 'A single neuron' and click *OK*.

All models in Nengo are built inside 'networks'. Inside a network you can put more networks, and you can connect networks to each other. You can also put other objects inside of networks, such as neural populations (which have single neurons inside of them), and input and outputs which go to and from those neural populations. For this model, you first need to create a neural population.

- Right-click inside the network you created and select *Create new->NEFEnsemble.* In the dialog box that appears, you can set *Name* to 'neuron', *Number of nodes* to 1, *Dimensions* to 1, *Node Factory* to 'LIF Neuron', and *Radius* to 1. Click *OK*.

A single neuron is now created. This leaky integrate-and-fire (LIF) neuron is a simple, standard model of a spiking single neuron. It resides inside a neural 'population', even though there is only one neuron. To readjust your view, you can zoom using the scroll wheel and drag the background to shift within the plane.

- To see neuron you created double-click on the 'population' you just created.

This shows a single neuron called 'node0'. To get details about this neuron, you can right-click it and select *Configure* and look through the parameters, though they won't mean much without a good understanding of single cell models. In order to simulate input this neuron, you need to add another object to the model that generates that input.

- Close the *NEFEnsemble Viewer* (with the single node in it) by clicking the 'X' in the upper right of that window.

- Right-click in the *Network Viewer* window and select *Create new->Function Input*. Set *Name* to 'input', and *Output Dimensions* to 1. Click *Set Functions*. In the drop down select *Constant Function*. Click *Set*. Set *Value* to 0.5. Click *OK*. Click *OK*. Click *OK*.

The object that will create a constant current to inject into the neuron has now been created. It has a single output labeled 'origin'. To put that into the neuron, you need to create a 'termination' (i.e., input) on the neuron.

- Right-click the 'neuron' population and select *Add decoded termination*. Set *Name* to 'input', *Weights Input Dim* to 1, and *tauPSC* to 0.02. Click *Set Weights*, double-click the value and set it to 1. Click *OK*.

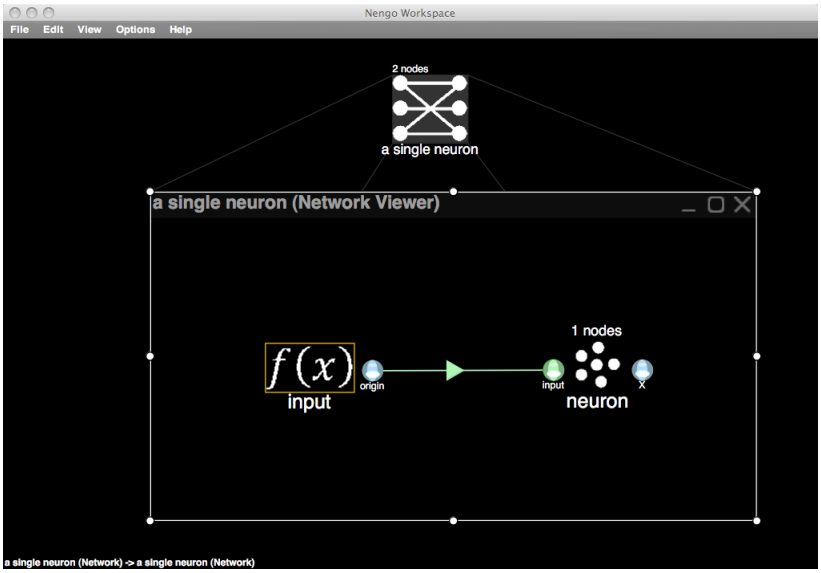

Figure 1.2: A single neuron Nengo model. This is a screen capture from Nengo that shows the finished model described in this section.

This takes the input and puts it directly into the neuron without changing the value. The *tauPSC* is a name for the 'synaptic time constant', which I will discuss in the next chapter. It is measured in seconds, so 0.02 is equivalent to 20 milliseconds. A new element will appear on the left side of the population. This is where you can hook the input function to.

- Click and drag the 'origin' on the input function you created to the 'input' on the neuron population you created.

Congratulations, you've constructed your first neural model. Your screen should look like Figure 1.2.

Running a model

You want to make sure your model works, so you need to use to the parts of Nengo that let you 'run' the model. There are two ways to run models in Nengo. The first is a 'non-interactive mode', which lets you put 'probes' into parts of the network. You then run the network for some length of time, and those probes gather data (such as when neurons are active, what their internal currents and voltages are, etc.). You can then view that information later using the built in data

viewer or with another program such as Matlab®. The other way to run models is in 'interactive mode', which is more hands-on, so I will use it in most of the examples in this book. An example of the running output in interactive mode for this model is shown in Figure 1.3.

- Right-click on the *Network Viewer* background and select *Interactive Plots*.

This pulls up a new window with a simplified version of the model you made. It has a 'play' button and other controls that let you run your model. First, you need to show some of the information generated by the model.

- Right-click on the 'neuron' population and select *spike raster*.

A spike raster shows the times that the neurons in the population fire an action potential spike (signifying a rapid change in its membrane voltage), which is largely how neurons communicate with one another. You can drag the background, and any of the objects around within this view to arrange them in a useful manner.

- Right-click on the 'neuron' population and select *value*.

This graph will show what effects this neuron will have on the current going into a neuron that receives its output spikes shown in the raster. Each small angular pulse is called a post-synaptic current or PSC. This is the current that is caused in the neuron after the synapse by this neuron's spike.

- Right-click 'input' and select *control*.
- Right-click on 'input' and select *value*.

This shows the value of your control input over time.

- Click the play arrow in the bottom right.

The simulation is now running. Because the neuron that you generated was randomly chosen, it may or may not be active with the given input. Either way, you should grab the slider control and move it up and down to see the effects of increasing or decreasing input. Your neuron will either fire faster with more input (an 'on' neuron) or it will fire faster with less input (an 'off' neuron). Figure 1.3 shows an on neuron with a fairly high firing threshold (just under 0.69 units of input). All neurons have an input threshold below (or above for off neurons) which they will not fire any spikes.

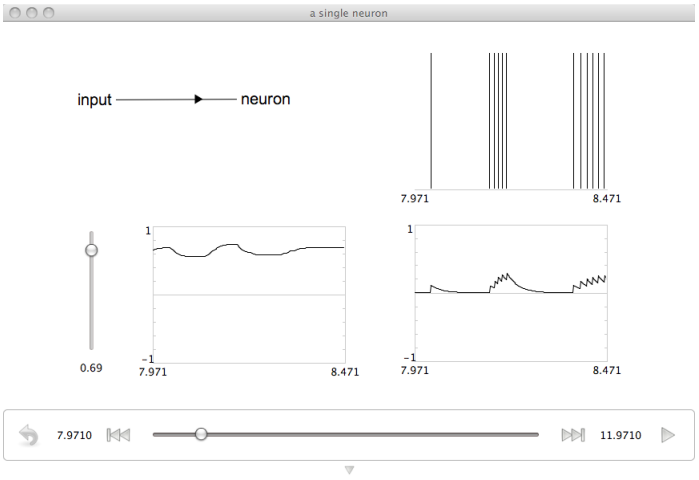

Figure 1.3: Running the single neuron model. This is a screen capture from the interactive plot mode of Nengo running a single neuron.

You can test the effects of the input and see if you have an on or off neuron, and where the threshold is. You can change the neuron by pausing the simulation, and returning to the main Nengo window. To randomly pick another neuron, do the following:

- Right-click on the 'neuron' population and select *Configure*.
- Click on the grey rightward pointing arrow that is beside *i neurons (int)*. It will point down and *i 1* will appear.
- Double-click on the 1 (it will highlight with blue), and hit *Enter*. Click *Done*.

You can now return to the interactive plots and run your new neuron by hitting play. Different neurons have different firing thresholds. As well, some are also more responsive to the input than others. They are said to have higher sensitivity, or 'gain'. You can also try variations on this tutorial by using different neuron models. Simply create another population with a single neuron and choose something other than 'LIF neuron' from the drop down menu. If you would like the neuron to spike less regularly, you can add a noise generator to your LIF neuron.[11]

[11]To do this, open the population, right-click the neuron you are using and select *Configure*. Under *noise* right-click *EMPTY* and select *Replace*. Under *PDF* right-click *EMPTY* and select

Congratulations, you have now built and run your first biologically plausible neural simulation using Nengo. You can save and reload these simulations using the *File* menu.

Replace. Select *GaussianPDF* and set the variance to 1.0. Click *Create,* then click *OK,* then set *frequency* to 1000. Then click *Create,* then click *OK,* then click *Done.* You can increase the noise by increasing the *variance* in the *Configure->noise->NoiseImplPDF->PDF* object.