1

Introduction

1.1 For whom is this book?

This book is for undergraduates and beginning graduate students. Of course, other graduate students, postdoctoral students, and anyone concerned with the fate of Western Civilization should read this book as well. Why focus on undergraduates? Because the malleable mind of the undifferentiated stem-person of college age is likely to be capable of making conceptual leaps that more ossified brains cannot. A theme that will recur throughout the book is the difficulty of unifying the compute and the neuro.

There are two major barriers to grand unification. First, computer science (like math, physics, and engineering) is made up of grand unifiers and their all-encompassing schemes, while neuroscience (like the rest of biology) is boatloads of facts. The differentiated mind of the engineer cannot swallow so many facts without a unifying framework. The differentiated mind of the biologist knows too much and is distracted by the many facts that contradict any preliminary framework someone tries to build. The fabled undergraduate mind, however, is notoriously unburdened by facts and yet willing to accommodate them. It (that mind, whether he or she) also typically seeks big pictures and is willing to take the leaps of faith required to acquire them.

The second barrier: these are early days. The field is a newly emerging hybrid and is itself still undifferentiated (an undifferentiated field needs an undifferentiated mind). It is a field still driven more by passion and fashion than by cool reason, emerging like a star from a gaseous cloud (OK that's a bit fanciful). Anyway, some of my colleagues will read this book and say that I've missed the whole point of the field. I've said the same of their books. Specifically, many researchers in the field come from the aforementioned engineering, physics, and math tradition. Their efforts are directed at developing a theoretical framework and their work tends to be colored by the framework they have chosen. I, on the other hand, come from the bio side. As a consequence, this book is a largely atheoretical approach to a theoretical science. I present stuff that is either fun or interesting or important and maybe sometimes all three.

In this computational neuroscience funhouse, I have included biological facts, computer science facts, equations, and theories both true and false. In some cases, I have included false theories because I don't know that they're false, though I may suspect it. In other cases, I introduce unlikely hypotheses just to roll them around and play with them. This interplay of facts and ideas makes up much of the work (or play) of modeling. Through experiencing it, the student will get a better idea of how and why modeling is done. In the last chapter, in particular, I bring the reader to my own particular circle of purgatory — the hall of perpetual mystification. I've tried to illustrate the complexities and contradictions of the field without unduly confusing the reader.

1.2 What is in the book?

Although my target undergraduate students are pluripotential, they are not yet omniscient. Specifically, they are majoring in philosophy, physics, math, engineering, biology, zoology, psychology, physical education, or business administration. As a result they know a lot about some things and little or nothing at all about others. For this reason, I have tried to cover all the basic bases. I have relegated much of this to a final chapter (Chap. 16), which can be read piecemeal as needed. In addition, many of the fundamentals have seeped into the main text as well. I have included a lot of basic computer science since an understanding of computational neuroscience requires a fairly sophisticated working knowledge of computers. This broad, blanket coverage means that certain chapters may seem trivial and others overly demanding for the particular student.

For the nonmathematical reader, the biggest challenge will be the advanced math topics, notably matrix algebra and numerical calculus. Though these topics are hard and can easily fill up a year of classroom instruction, I have tried to extract just the parts needed for present purposes and to make this accessible to anyone with mastery of high-school algebra. I have written out most equations in words so as to make them more accessible to anyone who is allergic to math. Additionally, the computer is a great leveler in this regard. Many once abstruse concepts in mathematics can now be quickly illustrated graphically. Computer in hand, this book can be enjoyed as a lighthearted romp through calculus, electrical engineering, matrix algebra, and other sometimes-intimidating topics.

For the nonbiological reader, the biggest challenge will be the profusion of facts and jargon words. This onslaught of information can be intimidating and discouraging. There is so much to know that it can be hard to know where to start. Often it is impossible to connect one set of facts to another set of facts. That is the goal of computational neuroscience. When you first encounter these facts, it will be without the benefit of such a model.

Although I have tried to write clearly and comprehensibly, I have also tried to use a lot of jargon. This can be annoying. I try to use jargon kindly and responsibly, to help the reader learn the words needed to read and converse knowledgeably in the many subfields that make up computational neuroscience. I have tried to always define jargon words immediately upon use in the text. As further assistance, I've provided a glossary. In addition to introducing jargon words, I also introduce some jargon concepts — touchstone ideas that are frequently referenced by people in a particular field. Having so much to introduce, concepts and phrases are sometimes mentioned, but not followed up on. They are presented to provide the reader with vocabulary and mental reference points for further reading or just plain thinking.

1.3 Do I need a computer for this book?

This book is meant to be read independently of any computer work. I have not put explicit exercises in the book but have made them available online (see below). One of the neat things about computational neuroscience is that it is so readily accessible. It is hard to get hold of the particle accelerators, centrifuges, and chimpanzees needed for most scientific study. But computers are everywhere, making computer-based research accessible to undergraduates and even to nonacademic folks. This is more true of computational neuroscience than it is of other computer modeling fields. To do weather prediction you need a supercomputer. A simple desktop PC will do for most of the material in this book. If the first stage of learning a field is to talk the talk by learning vocabulary, running the computer exercises will enable you to walk the walk as well.

Software

All of the figures in this book were put together using Neuron, a computer simulation program written by Mike Hines at Yale University. This program is freely available at *http://www.neuron.yale.edu*. Although Neuron is primarily designed for running the type of realistic simulations highlighted in the latter part of the book, it is flexible enough that I was able to use it for all the other simulations as well.

Software to produce all of the simulations and to run the emulator of Chap. 5 is available at these sites:

http://www.springer-ny.com/computer2brain
http://www.cnl.salk.edu/fctb
http://www.neuron.yale.edu/fctb

I will be pleased to consider additions or augmentations to this software, particularly if the contributor has already coded them.

Examples in the software are primarily presented through a graphical user interface. The reader or teacher who is interested in pursuing or presenting the subject in depth will want to become familiar with the Neuron program and with HOC, the Neuron programming language. This will allow the programs to be manipulated more flexibly in order to look at different aspects of a particular modeling problem.

Many of the examples presented in this book could also be readily programmed up in Matlab or Mathematica, or in other simulation programs such as Genesis or PDP++.

1.4 Why learn this now?

The genetic code was cracked in the mid-20th century. The neural code will be cracked in the mid-21st. Genetic science has given way to its applications in biotechnology and bioengineering. Neuroscience is still up-and-coming, the next big thing. Furthermore, as genetic manipulations and basic neuroscience add more raw facts to the broth, the need for meaning, structure, and theory becomes greater and greater. Enough information is coming together that the next generation of computational neuroscientists will make the leap into understanding. That means grant money, prizes, and fancy dinners! (If the movies are a guide, it also means evil robots, mind control, and dystopia, but let's not ruin the moment.)

There's such a variety of things to learn about in computational neuroscience that the student is in the position of the proverbial kid in the confectionery: so many problems to work on; so many amazing facts and theories from so many interrelated fields. Of course, this profusion of riches can also be frustrating. One doesn't know which gaudy bauble to pick up first, and, having picked one and discovered that it is not quite gold, strong is the temptation to drop it and pick up one that seems gaudier still.

1.5 What is the subtext?

In an age of ubiquitous computers, any topic can be discussed in their context, as attested to by the recent publication of *From Computer to Stain: Dry Cleaning in a Digital Age,* the inspiration for the title of this book. However, computational dry-cleaning is still just dry-cleaning done with computers. In fact, most of computational biology is just biology done with computers. Computational neuroscience is a little different. The computer itself represents the state of our knowledge about how complex information processing devices like the brain might work. For this reason, I have covered more computer science than would usually show up in a neuroscience book.

Present computer science curricula generally emphasize sophisticated abstractions that pull one away from the machine. Similarly, a branch of computational neuroscience has concerned itself with finding general principles of neural computation and has shied away from the messy meat of the brain. My contention is that the meat is the message for the brain and for the computer. Lovely abstract theories must grow out of an understanding of the machine. The most useful theories will be different for different machines. If there is a grand unified theory, it will stand abstract and austere away from the daily marketplace of synapses or transistors.

For this reason I have gone into some detail about the design and operation of an ancient computer, the PDP-8, a machine with the power of a modern pocket calculator. Such a simple machine can be readily described in a chapter. It also is small enough that one quickly runs into its limits and has to overcome them with programming tricks, commonly called "hacks." Hacking is now frowned upon in computer science, since it is mostly used to break into other people's machines. Biological evolution is one long history of hacking — using a piece of machinery for a new purpose and gradually working it into shape so that it seems to have been engineered from scratch for that purpose.

Understanding the process and products of evolution means understanding the problems of engineering with limited resources (and unlimited time). Programming a PDP-8 or rebuilding a diesel engine with pieces of scrap in a Third-World country requires ingenuity, ability to compromise, and willingness to make mistakes and start again. This process may leave us with a program or an engine with vestigial organs, tangled distribution routes, and inefficient procedures. Just as building the machine was a study in frustration, so examination of the machine will be a frustrating study that will also lead to dead-ends and back-tracking.

In this book, I repeatedly contrast my emphasis on the brain with the tendency of others in the field to focus on theory rather than detail. Perhaps I occasionally disparage these poor theoretical guys as cyborgs and hedgehogs. This is all in fun. Integration of theory and fact is a necessary goal in computational neuroscience. I try to give both their due in this book but I have not been successful in integrating them. The section titles — Computers, Cybernetics, and Brains — demonstrate this.

Our brains are full of contradictions but we learn to live with them. If we want to study the brain, we must be prepared for the kinds of ambiguities and occasional false leads that characterize life with our own brains.

1.6 How is the book organized?

Computational neuroscience is a new field whose essential paradigms are still the subject of debate. For this reason, it is not possible to present the basic material with the conceptual coherence of an introduction to well-established fields like chemistry or physics. The field remains a hodgepodge of exciting ideas and remarkable facts, some of which cannot be neatly conjoined. Instead of progress in a neat sequence from one idea to another, this book will at times seem to jump back and forth from one thing to another. This is an inherent difficulty of trying to teach both the computational and biological approaches in a single text. In general, I try to fill in the gaps where they can be filled in and point them out where they remain unbridged.

The organization of the book is as follows. I start with a brief introduction to neuroscience, touching on many but not all of the subfields that may need to be considered. I then go top-down with a description of how computers work. I start by noting how computers represent information. I then go into still more detail about the bits-and-bytes level of computer function. From there, I switch from transistors to neural network units and explore the concepts of artificial neural networks. This will entail a comparison between transistors and neurons and an explanation of how the artificial neural network units represent a compromise position between the two. From there, I show how these units can be connected together into artificial neural networks. I further expand on the artificial neural network paradigm, showing the use of these networks to explain the retina of a simple sea creature, the horseshoe crab. I look at another simple brain system, but this time one found in humans, the brainstem reflex that stabilizes the eyes in the head when the head is moved. Then, in a more speculative vein, I go still higher in the brain, looking at how artificial neural networks can be used to emulate aspects of human memory. This will involve an explicit compare-and-contrast with computer memory design.

Following this, I turn bottom-up, more seriously exploring the biological concepts of nervous system function. I start with a detailed description of the neuron with some ideas of how the different parts of the neuron can be modeled. I then explore in greater detail the two major techniques of realistic neuronal modeling: compartment modeling and the Hodgkin-Huxley equations. I then look at an example of how artificial neural network models of learning can inform our understanding of the brain and how study of the brain leads us to reconsider the details of these artificial neural networks.

The final chapter covers some details of the mathematical and scientific approaches and techniques used in this book. It includes unit analysis, binary arithmetic, linear algebra, calculus, and electronics. Comfort with handling units and scientific notation is needed for finding your way around science. Knowledge of binary is important for finding your way around a computer. Linear algebra is useful for finding your way around a network. Calculus is important for assessing movement and change. Electronics is needed for understanding electrical signaling in neurons. In each case, the material has been presented graphically and algebraically to make the subject accessible to those who do not feel comfortable with mathematical notation. It is expected that many readers will be unfamiliar with some or all of these areas and will want to read these sections as the technique comes up in the main text.

Since there are many topics touched on that do not always relate cleanly to one another, I tried to provide additional guidance. Each chapter begins with a brief introduction entitled “Why learn this?” Similarly, each chapter ends with “Summary and thoughts,” meant to synthesize concepts and remind the reader of what was learned. This section is “... and thoughts,” rather than “... and conclusions” because in many cases the conclusions await.

Springer
http://www.springer.com/978-0-387-95528-5

From Computer to Brain
Foundations of Computational Neuroscience
Lytton, W.W.
2002, XX, 364 p., Hardcover
ISBN: 978-0-387-95528-5