

Michele GIUGLIANO

# Computational Neuroscience: A Beginner's Guide

A primer in Computational Neuroscience

December 22, 2024

Springer Nature



*To Maura, who has kept walking with me,  
and in memory of Professors Grattarola and  
Tagliasco, who showed me the path.*



## **Foreword**

To be written by a person other than the author or editor of the book.



# Preface

To be written.

Modena,

*Michele GIUGLIANO*





## **Acknowledgements**

Special thanks to Maura and Margherita for their unwavering support. I am also in debt to several generations of students I taught while in Antwerp (2008-2019), Trieste (2019-2023), and Modena (2024-), for their precious feedback.



## Declarations

**Competing Interests** The Author has no conflicts of interest to declare that are relevant to the content of this book.

**Ethics Approval** Those experimental traces, collected in the Author's lab and included here for illustrative purposes, were obtained in compliance with the EU Directive 2010/63, the Belgian Royal Decree of 29/5/2013, the Italian Decree 26 of 4/3/2014, and upon authorisation by institutional Ethical Committees and national regulatory authorities.



# Contents



## Acronyms

Lists of abbreviations, symbols and the like are easily formatted with the help of the Springer-enhanced `description` environment.

ABC	Spelled-out abbreviation and definition
BABI	Spelled-out abbreviation and definition
CABR	Spelled-out abbreviation and definition





**Part I**  
**Elements of Computational Neurobiology**

Use the template *part.tex* together with the document class SVMono (monograph-type books) or SVMult (edited books) to style your part title page and, if desired, a short introductory text (maximum one page) on its verso page.

# **Chapter 1**

## **Electrical Phenomenology of Neurons**



## **Chapter 2**

# **Membranes and their Biophysics**



## Chapter 3

# Excitability and How to Model It

**Abstract** This chapter presents the basic principles of excitability, under the perspective of how it may be described and explored with models. I will be mostly referring to the generation of the action potential, which is an electrical pulse that travels along the membrane of excitable cells in the nervous system, muscles, heart, and the endocrine system. The way these pulses are ignited, propagated through space, and transformed is a fascinating outcome of natural evolution, emerged as a rapid communication mean across cells and tissues, as well as at the subcellular level.

### 3.1 Introduction

To be written. . .

### 3.2 Reduced Models of Excitability

#### 3.2.1 "As simple as possible, but not simpler"

This title is a portion of a famous quote, by Albert Einstein. Besides scientists, I have been always fascinated by the insights of the philosophers (of Science). Particularly, I found remarkable Karl Popper's recommendation for any scientific theory: it must be easily falsifiable if one wants it to have higher chances to be correct. Today this concept is in the DNA of researchers across many disciplines. Popper exemplified his point criticizing Astrology, for its large number and variety of presumed zodiacal influences that make it likely to account *a posteriori* for almost any outcome of human everyday life events. However, the very same inherent complications makes it very hard to identify a scenario to challenge Astrology's predictions!

A similar feeling is commonly embraced by physicists, who almost always first initiate a theoretical inquiries with a minimal description and with the simplest possible scenario of a physical system. Also data scientists and statisticians have the same concern, and rephrase it in terms of the risk of overfitting for a model with too many degrees of freedom when reproducing a dataset. If this happens, it is unlikely that the model will generalize to data it was never trained on.

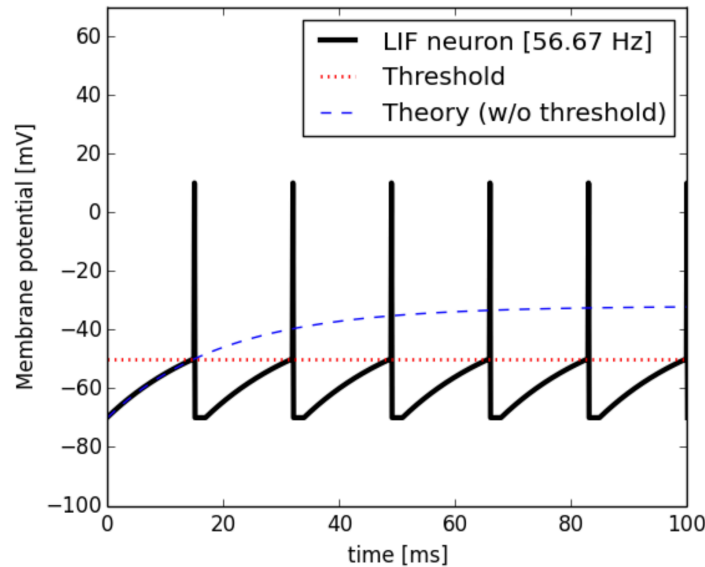
In Computational Neuroscience, we should then always be mindful when working with a very complicated description of a biophysical phenomenon, and when a very large number of parameters are a requirement of a model, even beyond the feasibility of the identification of all its parameters in *ad hoc* experiments. Hopefully, we should make our models as simple as possible, and we should set the condition for its predictions to be falsified: that is when we learn, in a virtual interaction between experiments and theory.

There is however another important consideration I would like to do. This originally escaped my own deep understanding and appreciation when I was a young student. At that time, after learning about the Hodgkin and Huxley seminal work I felt empowered to imagine its arbitrary extension *in principle*: any number of additional ionic currents could be included using Markov kinetic schemes, a more accurate description of cell's morphology included by cable theory, electrodiffusion for a variety of ionic concentrations coupled with the other equations, and so on. As computers were used to solve the model equations and perform numerical simulations, I thought that only the sky was the limit. My optimism was perhaps only partly correct. As for weather forecast, I imagined that large scale supercomputers could *crunch numbers* and solve a huge number of equations, describing a very complex (weather) model, and ultimately produce predictions for atmospheric physical variables in the immediate and near future. However, even assuming that all the equations and parameters are known and do capture accurately the underlying physical reality, there are two problems: 1) *number crunching* alone requires time to be completed, sometimes a substantial amount of time; 2) simulating a model on a digital computer requires iteratively feeding partial numerical results into a series of mathematical expressions, and errors may accumulate. Both issues come as computers are intrinsically discrete *in time* and limited in the resolution for numerical representation is limited, so that rounding errors or imperfect numerical processing might decrease our confidence over model behavior as the simulated time goes by.

These issues become in my opinion problems, as the the intuition and insights that a computational neuroscientist gains over time, when developing, testing, and exploring their model might be challenged. The time one has to wait for a simulation to complete does unavoidably makes it harder for intuition to develop on how, e.g., changing one parameter affects the subtle behavior of a certain phenomenon.

In addition, models so rich and complicated might further make it hard for the scientist to recognize invariants or key mechanisms and principles, as a consequence of the likely mathematical intractability, in terms of derivation of general results.





**Fig. 3.1** Please write your figure caption here

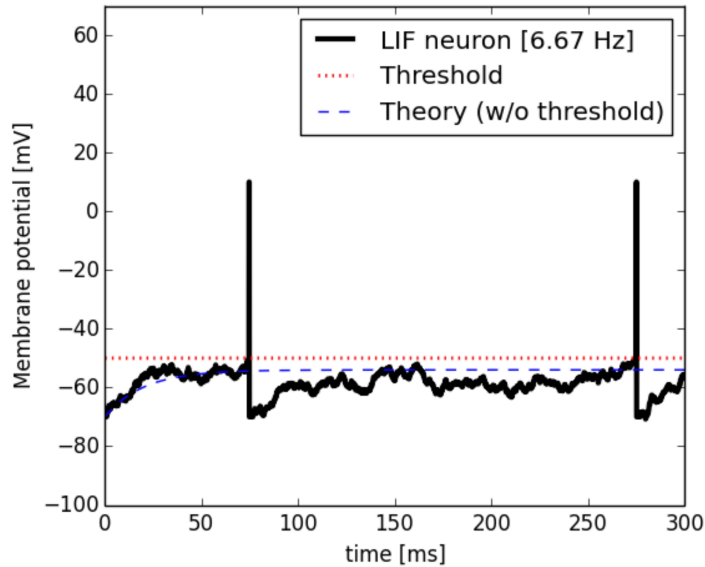
### 3.2.2 From conductance-based models to Integrate-and-Fire models

Those who are mathematically inclined, may appreciate a very early elegant presentation, by Larry Abbott and Kepler (??), deriving integrate-and-fire models equations from conductance-based descriptions. On a first approximation, you are separating time scales associated to the many state variable of a neuron model. Focusing for instance only on the role of ionic conductances contributing to the initiation of an action potential, a similar procedure rewrites the charge-balance equation.

#### 3.2.3 The linear Integrate-and-Fire model

When considering neuronal activation, some variables operate much faster than others, potentially reaching a steady state instantaneously. Take, for example, the 'm' variable and its associated function, 'm infinity,' which represents this asymptotic state. One could simplify the model by focusing solely on this rapid activation and using fewer parameters to describe action potential initiation. However, such a simplified model might fail to capture the complexities of neuronal activity, particularly the spatial dynamics of signal conduction.

A more practical approach to defining 'integrate-and-fire' neurons stems from recognizing that the action potential itself is a highly stereotyped event. Its amplitude carries little information; therefore, as a first approximation, it's reasonable to



**Fig. 3.2** Please write your figure caption here

conserve computational resources and avoid detailed analysis of the action potential's exact trajectory. This perspective leads to a simplified model with only two main variables: the membrane potential and the integrated input current. This simplification is particularly relevant for cortical neurons, which exhibit sparse activity, firing action potentials only briefly before returning to a resting state. Consequently, we can describe the membrane potential's evolution up to the point of action potential firing and then resume the description after the action potential concludes.

Several approaches exist to describe the equation governing this simplified model. The simplest, known as the 'linear integrate-and-fire with threshold' model, treats the neuron as a perfect integrator. The membrane potential's derivative is directly proportional to the input current. This simplification disregards the subthreshold influence of individual voltage-gated ion channels, which contribute to neuronal integration. However, for exploratory purposes, focusing on this simplified phenomenon can be valuable.

It's important to highlight that this model adopts a 'point neuron' perspective. Most integrate-and-fire models aim to describe the electrical activity of neurons as a single point unit, effectively ignoring the complexities of dendritic integration. This simplification can be considered a temporary measure, with the complexities of dendritic processing potentially addressed in more advanced models.

By accepting this point neuron approximation, we can describe the electrical behavior of a neuron using a limited number of state variables, typically just the membrane potential, which varies over time without spatial considerations. However, it's crucial to acknowledge that this perfect integrator model has limitations. It doesn't

accurately reflect the membrane potential's response to inhibitory and excitatory inputs, as it would unrealistically accumulate charge without returning to the resting membrane potential.

Let's add a constant parameter, potentially a negative number, to the right-hand side of our equation. This introduces a balance between the input strength and a 'pulling force' that determines the decay rate of the membrane potential. Consequently, the membrane potential avoids indefinite accumulation and instead decays towards a resting value in the absence of synaptic input.

However, without any constraints, the membrane potential could theoretically approach negative infinity. To prevent this, especially when synaptic activity isn't consistently excitatory, we introduce a lower boundary or 'floor.' This acts as a clipping mechanism, preventing the membrane potential from decreasing beyond a certain hyperpolarized value.

With this addition, our model's equation becomes:  $dV/dt = -\beta + I_{\text{sub}_\ell \text{ext}_i / \text{sub}_\ell}$ , where ' $\beta$ ' represents the decay rate and  $I_{\text{sub}_\ell \text{ext}_i / \text{sub}_\ell}$  represents the external current, which could be synaptic in nature. This simple model highlights the interplay between capacitance, decay rate, and threshold in shaping neuronal integration.

While this model doesn't fully capture the complexities of real neurons, its simplicity offers valuable insights. For instance, we can explore the concept of a fixed action potential threshold. Although a precise threshold isn't feasible in real neurons or even in detailed conductance-based models, assuming a fixed threshold allows us to define a firing mechanism for our model.

When the membrane potential exceeds this threshold, the neuron is said to have fired an action potential, and the membrane potential is instantly reset to a lower value. This reset value, typically above the resting potential, can reflect the after-spike hyperpolarization observed in real neurons.

Furthermore, we can incorporate a refractory period, a brief period following an action potential where the neuron cannot integrate new inputs. This phenomenologically captures the transient inactivation of ion channels observed in real neurons.

This simple model allows us to derive analytical relationships between input current and output firing frequency. By considering the absolute refractory period, we can even determine a maximum firing frequency, an upper bound that wouldn't be easily attainable with more complex models.

Analyzing the f-I curve, a plot of firing frequency versus input current, is a standard experimental protocol for characterizing neuron types. In our model, a negative current results in no spikes, as the membrane potential continuously decreases until it reaches the lower boundary.

For spiking to occur, the input current must exceed a minimal value known as the rheobase. If the current equals the decay rate ( $\beta$ ), the membrane potential remains constant. When the current surpasses  $\beta$ , the membrane potential increases and eventually reaches the threshold.

By solving the differential equation, we can calculate the time it takes for the membrane potential to reach the threshold from a given starting point. This interspike interval is constant under constant input, matching the behavior of some real neurons and allowing us to calculate the average firing frequency.

Despite its simplicity, this model provides insights into the roles of capacitance and decay in neuronal firing. We can even extend the model to consider random, Poisson-distributed input arrival times, simulating fluctuating synaptic inputs. This leads to a stochastic process for the membrane potential, offering a simplified description of certain neuronal regimes.

In these lectures, we'll simplify the complexities of neuronal excitability we've discussed in previous weeks. Last week, we explored the Hodgkin-Huxley model, which uses the charge balance equation to describe neuronal dynamics. This model incorporates various ion channels, such as sodium and potassium channels, and their associated conductances and gating variables. These variables, like 'm' and 'h' in the Hodgkin-Huxley model, influence the flow of ions across the membrane, shaping the action potential.

Today, we'll focus on a simpler model called the integrate-and-fire model. This model capitalizes on the stereotypical nature of action potentials, simplifying the complex dynamics of ion channels. While the Hodgkin-Huxley model requires computationally intensive simulations, the integrate-and-fire model allows for more efficient simulations and even pen-and-paper calculations.

One key aspect we'll explore is the relationship between input current and firing frequency, represented by the f-I curve. This curve illustrates how a neuron's firing rate changes in response to varying input currents. With the integrate-and-fire model, we can derive an analytical expression for the f-I curve, providing insights into neuronal behavior without relying on complex simulations.

To begin, let's revisit the passive membrane properties. We'll simplify the neuron to an RC circuit, where 'R' represents the membrane resistance and 'C' represents the membrane capacitance. This simplification disregards the complexities of voltage-gated ion channels, focusing solely on the passive flow of current across the membrane.

In the integrate-and-fire model, we introduce a fixed threshold. When the membrane potential reaches this threshold, we say the neuron has fired an action potential. This simplification disregards the intricate dynamics of ion channel activation and inactivation that shape the action potential in real neurons.

After firing, the membrane potential is instantly reset to a lower value and held there for a brief refractory period. This mimics the refractory period observed in real neurons, where they are temporarily unable to fire another action potential.

This simplified model, with its fixed threshold and reset mechanism, allows us to explore neuronal behavior under different input conditions. We can analyze how the membrane potential integrates input current and how the firing frequency changes with varying input strengths.

To make the model more realistic, we can incorporate additional features, such as adaptation. This reflects the phenomenon observed in real neurons where their firing rate decreases over time in response to a constant input. We can achieve this by adding an adaptation current that acts as a negative feedback mechanism, reducing the firing frequency as the neuron becomes more active.

One way to understand adaptation is by considering the role of calcium ions. During an action potential, calcium ions flow into the neuron, and their concentration

increases. This increase can activate calcium-dependent ion channels, which in turn can hyperpolarize the neuron, reducing its excitability and firing rate.

By incorporating calcium dynamics into the integrate-and-fire model, we can capture the phenomenon of spike frequency adaptation. This allows the model to more accurately reflect the behavior of real neurons, which exhibit a decrease in firing rate over time in response to a constant input.

While the integrate-and-fire model is a simplification, it offers valuable insights into neuronal behavior. It allows us to explore fundamental principles of neuronal integration and firing, and it can be extended to incorporate additional complexities, such as adaptation and calcium dynamics. By understanding the strengths and limitations of this model, we can gain a deeper appreciation for the intricate workings of real neurons and their role in shaping brain function.

To illustrate the usefulness of the integrate-and-fire model, let's examine some experimental data. We'll compare the f-I curves of real neurons with those predicted by the model. In these experiments, we recorded from different types of neurons in the somatosensory cortex of rats, including pyramidal cells and interneurons.

By injecting varying levels of current into these neurons, we observed how their firing rates changed. We then compared these experimental f-I curves with those generated by the integrate-and-fire model, adjusting the model parameters to best fit the data.

The results showed a remarkable agreement between the experimental and model-generated f-I curves. This suggests that the integrate-and-fire model, despite its simplicity, can effectively capture the firing behavior of real neurons. This is particularly valuable because it allows us to explore neuronal dynamics without resorting to complex and computationally expensive models like the Hodgkin-Huxley model.

However, it's important to acknowledge the limitations of the integrate-and-fire model. One key limitation is its inability to accurately predict the precise timing of action potentials. This is because the model's fixed threshold mechanism doesn't fully capture the dynamic interplay of ion channels that shape the action potential in real neurons.

To address this limitation, alternative models have been developed, such as the exponential integrate-and-fire model. This model incorporates a more realistic threshold mechanism that accounts for the explosive activation of sodium channels near the threshold. By doing so, it can more accurately predict the timing of action potentials, especially in response to fluctuating inputs.

The integrate-and-fire model, despite its limitations, remains a valuable tool in neuroscience research. Its simplicity allows for efficient simulations and analytical calculations, providing insights into neuronal behavior and network dynamics. By understanding its strengths and limitations, we can effectively utilize this model to explore the complexities of brain function and develop a deeper appreciation for the intricate workings of neurons.

In addition to its applications in single neuron modeling, the integrate-and-fire model can also be extended to simulate networks of interconnected neurons. This allows us to explore how populations of neurons interact and generate complex

dynamics, offering insights into network-level phenomena such as oscillations and synchronization.

Imagine a network of integrate-and-fire neurons connected by synapses. When a neuron fires an action potential, it sends a signal to its downstream synaptic partners. This signal can either excite or inhibit the partner neuron, influencing its likelihood of firing.

By simulating such networks, we can investigate how different connectivity patterns and synaptic properties shape network activity. This can help us understand how the brain processes information and generates complex behaviors.

Furthermore, the integrate-and-fire model can be adapted to incorporate various levels of realism. For instance, we can include different types of neurons with distinct firing properties, or we can model the spatial structure of neurons, considering the dendritic and axonal compartments.

These extensions allow us to explore a wider range of neuronal phenomena and network dynamics. For example, we can investigate how different neuron types contribute to network oscillations, or how dendritic integration influences neuronal firing.

Overall, the integrate-and-fire model provides a versatile framework for exploring neuronal and network dynamics. Its simplicity allows for efficient simulations and analytical calculations, while its flexibility enables the incorporation of various levels of realism. By utilizing this model effectively, we can gain a deeper understanding of the brain's intricate workings and unravel the complexities of neural information processing.

It's important to remember that the integrate-and-fire model, while a powerful tool, is still a simplification of real neuronal behavior. It captures essential features of neuronal dynamics, but it disregards many intricate details.

For instance, the model assumes a fixed threshold for action potential generation, whereas real neurons exhibit more complex threshold dynamics due to the interplay of various ion channels. Additionally, the model typically represents neurons as point units, neglecting their spatial structure and the complexities of dendritic integration.

Despite these simplifications, the integrate-and-fire model has proven remarkably successful in capturing a wide range of neuronal phenomena. It has been used to study single neuron behavior, network dynamics, and even cognitive processes.

Moreover, the model's simplicity allows for analytical tractability, enabling the derivation of mathematical relationships that provide insights into neuronal function. This analytical tractability is a major advantage over more complex models, which often require computationally intensive simulations.

As we conclude our exploration of the integrate-and-fire model, it's crucial to recognize that it represents just one member of a broader family of simplified neuron models. Each model within this family offers a unique balance between simplicity and biological realism.

For instance, some models emphasize mathematical tractability, allowing for analytical solutions and insights into fundamental principles. Others prioritize biological accuracy, incorporating more detailed representations of ion channels and dendritic integration.

The choice of which model to use depends on the specific research question and the desired level of detail. For studies focused on network dynamics, a simpler model like the integrate-and-fire model might suffice. However, for investigations into the intricate dynamics of single neurons, a more detailed model might be necessary.

Researchers are continually developing new and improved neuron models, striving to strike an optimal balance between simplicity and realism. These models are essential tools for understanding the complexities of brain function and unraveling the mysteries of neural information processing.





## **Chapter 4**

# **Synapses and their Plasticity**



## **Chapter 5**

### **Simulating or Modeling networks?**



## **Chapter 6**

# **Dynamical Systems**



## Appendix A

### Chapter Heading

*All's well that ends well*

Use the template *appendix.tex* together with the Springer document class SVMono (monograph-type books) or SVMult (edited books) to style appendix of your book.

#### A.1 Section Heading

Instead of simply listing headings of different levels we recommend to let every heading be followed by at least a short passage of text. Furtheron please use the  $\LaTeX$  automatism for all your cross-references and citations.

##### A.1.1 Subsection Heading

Instead of simply listing headings of different levels we recommend to let every heading be followed by at least a short passage of text. Furtheron please use the  $\LaTeX$  automatism for all your cross-references and citations as has already been described in Sect. ??.

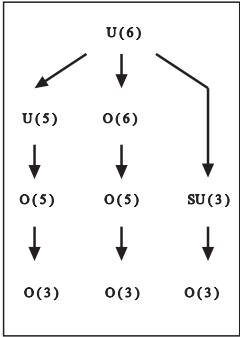
For multiline equations we recommend to use the `eqnarray` environment.

$$\begin{array}{l} \mathbf{a} \times \mathbf{b} = \mathbf{c} \\ \mathbf{a} \times \mathbf{b} = \mathbf{c} \end{array} \quad (\text{A.1})$$

##### A.1.1.1 Subsubsection Heading

Instead of simply listing headings of different levels we recommend to let every heading be followed by at least a short passage of text. Furtheron please use the  $\LaTeX$  automatism for all your cross-references and citations as has already been described in Sect. ??.

**Fig. A.1** Please write your figure caption here



Please note that the first line of text that follows a heading is not indented, whereas the first lines of all subsequent paragraphs are.

**Table A.1** Please write your table caption here

Classes	Subclass	Length	Action Mechanism
Translation	mRNA <sup>a</sup>	22 (19–25)	Translation repression, mRNA cleavage
Translation	mRNA cleavage	21	mRNA cleavage
Translation	mRNA	21–22	mRNA cleavage
Translation	mRNA	24–26	Histone and DNA Modification

<sup>a</sup> Table foot note (with superscript)



**Part II**  
**Networks of Spiking Neurons**

Use the template *part.tex* together with the document class SVMono (monograph-type books) or SVMult (edited books) to style your part title page and, if desired, a short introductory text (maximum one page) on its verso page.

# Glossary

Use the template *glossary.tex* together with the Springer document class SVMono (monograph-type books) or SVMult (edited books) to style your glossary in the Springer layout.

**glossary term** Write here the description of the glossary term. Write here the description of the glossary term. Write here the description of the glossary term.

**glossary term** Write here the description of the glossary term. Write here the description of the glossary term. Write here the description of the glossary term.

**glossary term** Write here the description of the glossary term. Write here the description of the glossary term. Write here the description of the glossary term.

**glossary term** Write here the description of the glossary term. Write here the description of the glossary term. Write here the description of the glossary term.

**glossary term** Write here the description of the glossary term. Write here the description of the glossary term. Write here the description of the glossary term.



# Solutions

## Problems of Chapter ??

?? The solution is revealed here.

### ?? Problem Heading

- (a) The solution of first part is revealed here.
- (b) The solution of second part is revealed here.