

Spike neural models

Part I: The Hodgkin-Huxley model

Melissa G. Johnson^a,  and Sylvain Chartier^a


^aUniversity of Ottawa

Abstract ■ Artificial neural networks, or ANNs, have grown a lot since their inception back in the 1940s. But no matter the changes, one of the most important components of neural networks is still the node, which represents the neuron. Within spiking neural networks, the node is especially important because it contains the functions and properties of neurons that are necessary for their network. One important aspect of neurons is the ionic flow which produces action potentials, or spikes. Forces of diffusion and electrostatic pressure work together with the physical properties of the cell to move ions around changing the cell membrane potential which ultimately produces the action potential. This tutorial reviews the Hodgkin-Huxley model and shows how it simulates the ionic flow of the giant squid axon via four differential equations. The model is implemented in Matlab using Euler's Method to approximate the differential equations. By using Euler's method, an extra parameter is created, the time step. This new parameter needs to be carefully considered or the results of the node may be impaired.

Keywords ■ Spiking neural networks, neural models, Hodgkin-Huxley model. **Tools** ■ Matlab.

Acting Editor ■ Denis Cousineau (Université d'Ottawa)

 mjohn140@uottawa.ca

 *MGJ*: na; *SC*: na

 [10.20982/tqmp.13.2.p105](https://doi.org/10.20982/tqmp.13.2.p105)

Introduction

The arrival of computers ushered in the advent of artificial neural networks (ANNs). ANNs have a variety of purposes, such as regression and classification (Dreiseitl & Ohno-Machado, 2002), learning (Sutskever, Vinyals, & Le, 2014), recognition (Taigman, Yang, Ranzato, & Wolf, 2014), prediction (Shen & Bax, 2013), and helping researchers understand the human brain (Kuebler & Thivierge, 2014). Over time, three distinct generations of ANNs have emerged: the first generation is digital, the second generation is analog, and the third generation is spiking.

The origin of ANNs is rooted in biology, and for these first generation networks the main biological feature they used was the all or nothing aspect of neural firing: to fire or not fire, 0 or 1. In a landmark paper by McCulloch and Pitts (1943), they use the “all-or-none” characteristic of neural activity and fashioned the idea of modelling neurons to be threshold gates which either fire or don't fire (McCulloch & Pitts, 1943). The general idea behind the threshold

gate is that the computational unit of the network receives input from other units in the form of a 1 (fire) or 0 (didn't fire) and sums up these inputs. If the summation passes a pre-determined threshold, the unit fires, or outputs 1, and if the summation does not pass the threshold, the unit outputs a 0 to signify not firing. This concept of a threshold gate for the nodes (or neurons) of the network gave rise to a number of ANNs including the perceptron (Rosenblatt, 1958) and the Hopfield network (Maass, 1997; Denker et al., 1987; Hopfield, 1982). These networks helped develop the idea of using neural networks to understand how information is stored and remembered (Rosenblatt, 1958).

Digital input and output has limited capabilities though; first generation ANNs cannot handle “degrees”, such as humans' ability to see degrees of colour or feel degrees of pressure through touch. The second generation of ANNs allows for analog input and output. This generation is much more robust and flexible, able to solve a wider range of problems and solve them more efficiently. For example, the second generation handles greyscale, instead of



just black and white, and also solves logic problems using fewer nodes (DasGupta & Schnitger, 1994).

Analog data works because of the addition of an activation function which is an abstraction of the neural firing rate. In second generation ANNs, the choice of an activation function is very important as different activation functions (e.g. sigmoid or cubic) differ in terms of efficiency and quality of approximation (DasGupta & Schnitger, 1994). Many feedforward, reinforced learning, and backpropagation networks are second generation neural networks.

Work on the second generation ANNs is still ongoing, both improving current ANNs and developing new ones. Applications that use deep learning, such as speech recognition and computer vision, are examples of second generation ANNs (LeCun, Bengio, & Hinton, 2015). In fact, the vast majority of applications that use ANNs employ second generation ANNs. Still, research is realizing that second generation ANNs cannot match the human brain's capabilities. For example, humans have extremely fast processing speeds and are very efficient for learning and recognizing items in their environment, especially considering the different forms and views objects can take. Therefore, the third generation returns to our biological roots and looks at how our brains function (Maass, 1997). Specifically, the third generation of ANNs are spiking neural networks (SNNs) where action potentials (spikes) form the input and output of the network nodes. This goes beyond the 'all or nothing' of the first generation because they include other important aspects of neurons such as how the spikes might encode information. The third generation of ANNs may also take into account different features of the neurons such as how much of the neurotransmitter is released, or the time delay between different presynaptic spikes. The added information from biologically inspired nodes gives more opportunities for information encoding and decoding.

In all generations though, the first step in understanding an ANN is understanding the node, or neuron. SNN nodes can be divided into three categories based on their similarity to biological neurons (Zamani, Sadeghian, & Chartier, 2010).

Explicit representations of biological neurons. These models are the most biologically accurate. Different parameters of the model represent specific biological components of a neuron. These models explain how neurons function in detail but they are computationally expensive and therefore simulations tend to be slow.

General representation of neural networks. While these models are not quite as biologically detailed as the previous models, they are still complex models composed of multiple differential equations. In this category, the pa-

rameters do not exactly correspond to a biological component; they model the biological behavior while ignoring the physical characteristics that make the behavior happen.

Generic threshold-fire neural networks. These models are the simplest in terms of equations and the farthest from accurately representing the biological neuron. They model a neuron's ability to integrate inputs and fire via a threshold.

The above categories are regarding the nodes only; the architecture of the SNNs can also vary in complexity and functionality. Trying to review all SNNs would require a complete book. Therefore, for a meaningful coverage of the subject, this tutorial primarily focuses on the node. This tutorial is split into two parts, part one focuses on the explicit representation of the neuron, specifically on the Hodgkin-Huxley model of the node while part two will focus on the other two categories of SNN nodes.

The node represents the neuron in the brain, or, more accurately, it represents the cell body (soma) of the neuron. Much like the cell body, the node receives signals from other nodes and decides if it is going to send a signal along to other nodes. No ANN, spiking or otherwise, exists without a node and in SNNs there are a lot of options to choose from. The Hodgkin-Huxley model is a point neuron model. Point neuron models are only concerned with how the neuron handles input voltage to produce, or not produce, an action potential. They are not concerned with more complex features of neurons that can affect the buildup and dissemination of the action potential. There are models that also incorporate other features, for example, the compartmental neuron model takes into consideration time of inputs: inputs come in from different sources in different locations therefore their signals take varying amounts of time to reach the summation point (Gerstner, 2002).

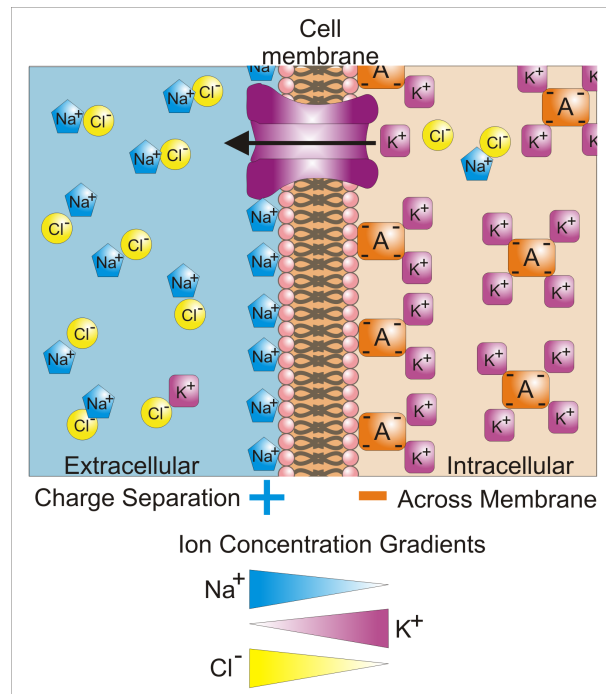
The node in the Hodgkin-Huxley model simulates the biological functioning of the neuron. Therefore before reviewing the model, a brief review of the biological neuron is in order. This biological review is followed by an explanation of the Hodgkin-Huxley model and then a discussion on how to implement the model (see Appendix A for Matlab code of the Hodgkin-Huxley model).

The Biological Neuron

The Hodgkin-Huxley model is an example of an explicit representation of the neuron. This model is concerned with how the movements of ions produce the changes in the voltage of the neuron. Therefore, to understand what this model mimics, a basic knowledge of the ionic changes is important. There are other models in the literature that focus on different aspects of the neuron.



Figure 1 ■ The ionic basis of the resting potential. Diagram create by Synaptitude at English Wikipedia and retrieved from commons.wikimedia.org/wiki/File:Basis_of_Membrane_Potential2.png.



Ionic Movement

When a neuron is at rest (no input current) intracellular fluid is negatively charged and has a high concentration of potassium ions (K^+) and organic ions (A^-) with smaller concentrations of chloride ions (Cl^-) and sodium ions (Na^+). The extracellular fluid is positively charged and contains high concentrations of Cl^- and Na^+ but low concentration of K^+ (see Figure 1). The cell membrane is semipermeable and separates the intracellular fluid from the extracellular fluid. The permeability of the cell membrane depends on the ion; it is more permeable to K^+ than it is to Na^+ , and it is not permeable to A^- . This permeability, the forces within the neuron, ion pumps, and the ionic channels control the movement of all the ions.

There are two forces working on the ions at all times: diffusion and electrostatic pressure. These forces affect ionic movement in the extra- and intracellular fluid. The force of diffusion moves ions so that each ion is equally spread throughout the fluid with no areas of high or low concentrations. Therefore, because there is a high concentration of K^+ in the intracellular fluid, diffusion exerts pressure to move some of the ions to the extracellular fluid. Diffusion also exerts pressure on the Na^+ ions to move from

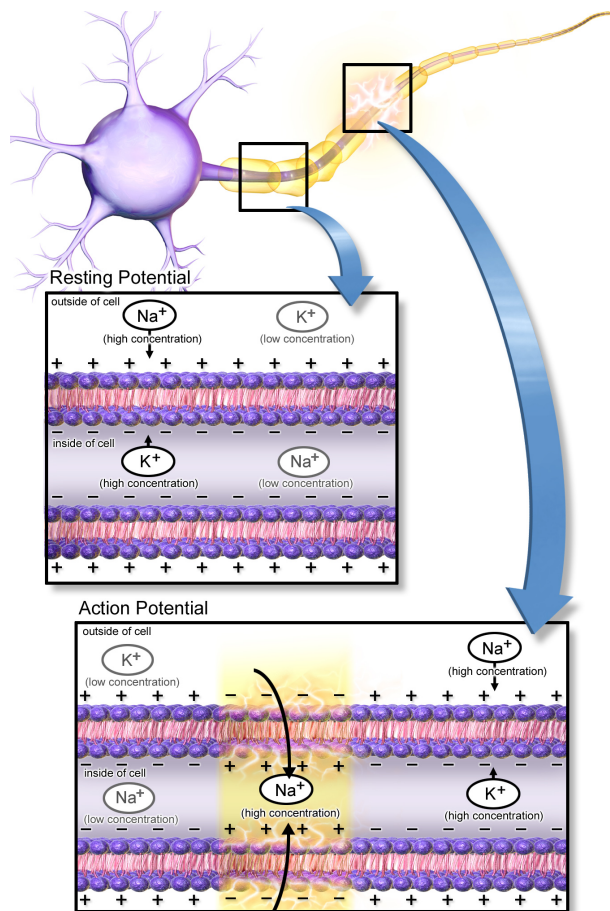
the extracellular fluid to the intracellular fluid. If there was just diffusion working on these ions, the neuron would have equal Na^+ and K^+ ions in the intra- and extracellular fluid since the force of diffusion breaks up any areas of high concentration of ions, moving the ions to areas with lower concentrations.

Electrostatic pressure causes ions of the same charge to be repulsed by each other while making ions of opposite charges attracted to each other. Extracellular fluid is positively charged therefore it repulses positively charged ions, such as K^+ , while attracting negatively charged ions such as Cl^- . Similarly, intracellular fluid is negatively charged therefore repulsing Cl^- while attracting K^+ .

Both K^+ and Cl^- have two forces exerting opposing pressures on them which holds their overall concentration stable within the intra- and extracellular fluid. For Na^+ ions, the force of diffusion and electrostatic pressure both exert pressure to move these ions from the extracellular fluid to the intracellular fluid. Therefore, to achieve stability, the cell membrane contains sodium-potassium pumps which pushes three Na^+ ions out of the cell in exchange for pumping two potassium ions into the cell. Overall, these pumps and the cell membrane permeability keep the balance of Na^+ and K^+ stable in the intra- and extracellular fluid



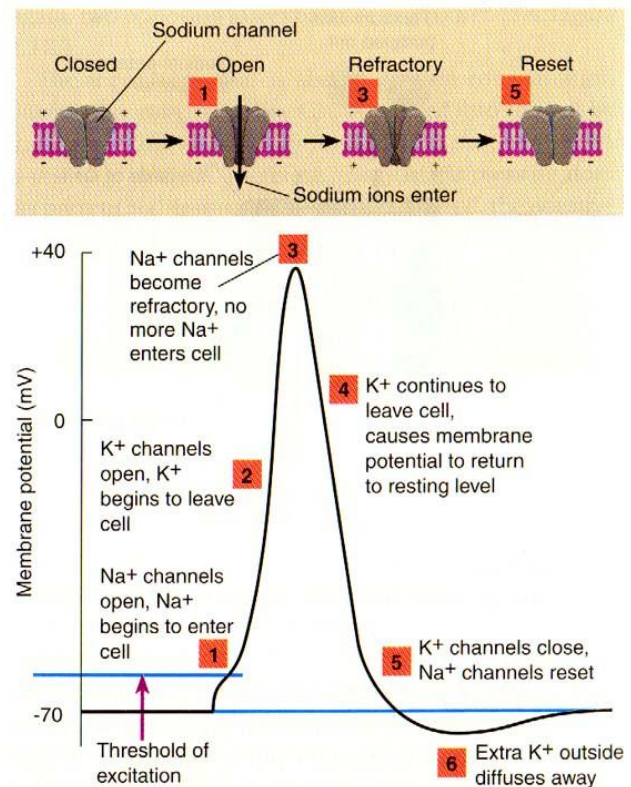
Figure 2 ■ Ion locations while at rest (top) compared to during an action potential (bottom). Diagram created by Blausen Medical Communications, Inc. and retrieved from commons.wikimedia.org/wiki/File:Blausen_0011_ActionPotential_Nerve.png.



despite extra pressure on the Na^+ to enter the intracellular fluid.

Cells also have different ion channels which, when open, allow ions of a particular type to flow through the cell membrane (see Figure 2). The opening and subsequent closing of these channels affect, and are affected by, the voltage of the cell membrane and are the cause of action potentials. When a current enters the cell it changes the membrane potential. An action potential, the rapid movement of ions, starts when the membrane potential reaches a prescribed threshold due to external sources of input current. When the membrane potential reaches said threshold, Na^+ channels open allowing Na^+ to enter the cell (flood the intracellular fluid) which causes the membrane potential to spike. K^+ also have channels to allow K^+ ions to leave the intracellular fluid but these channels require a higher membrane potential voltage than the Na^+ channels. There-

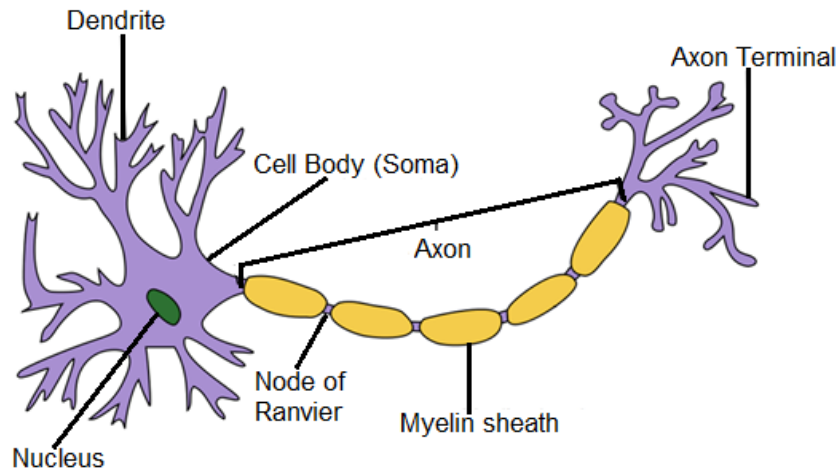
Figure 3 ■ Ion movements during an action potential. Bottom image shows when channels open or close during an action potential. Top image shows the corresponding sodium channel at specific points of the action potential. Diagram created by: If Only and retrieved from scioly.org/wiki/index.php/File:Image12.jpg.



fore, first Na^+ enters the cell and creates a sharp increase in the membrane potential. This increase then causes the K^+ channels to open, allowing K^+ to leave the intracellular fluid. At this point, Na^+ is entering faster than K^+ is leaving because of the electrostatic and diffusion forces exerted on it, so there is a still a net gain in membrane potential. When the peak occurs, Na^+ channels become refractory (blocked) so Na^+ can no longer enter the cell, but K^+ can still leave, thereby making the membrane potential decrease (see Figure 3). Once the resting potential of the membrane is reached K^+ channels close and Na^+ channels reset so that they can eventually be opened again. At the end of the action potential, the mix of Na^+ and K^+ in the intra- and extracellular fluid is not the stable mixture described earlier. Therefore the forces of diffusion and electrostatic pressure are not balanced, producing a refractory period where the cell membrane goes below the resting point before increasing again as the ions move into their stable states.



Figure 4 ■ General structure of a neuron. Diagram created by Quasar Jarosz and retrieved from https://commons.wikimedia.org/wiki/File:Neuron_Hand-tuned.svg. Edited to include labels, modified axon label, and removed Schwann cells.



Neuron Overview

Ion movements and the subsequent action potentials are just a small part of how neurons work. As mentioned before, there are other models that deal with other aspects of neurons and how the brain functions as a whole. Therefore, for a more thorough understanding of the complexity involved in biological modelling, below is a very quick and general review of how the brain works. There are also many resources for more detailed explanations, such as those by Carlson and Birkett (2016) or Dubois (2010).

As seen in Figure 4, most neurons contain a cell body (soma), dendrites in the form of dendritic tree(s) which contain dendritic spines at the ends, axon(s) which may be covered in multiple myelin sheaths (the nodes of Ranvier are areas between the myelin sheath), and terminal buttons at the axon terminal. The dendritic spines receive messages from other neurons while terminal buttons secrete a neurotransmitter that is turned into messages for other neurons. The soma contains the nucleus and sums up incoming currents from the dendrites. The configuration of these components can vary depending on the type of neuron. For example, a bipolar neuron has one dendritic tree, which has many dendrites on it. The one dendritic tree is connected to the soma. The soma is also connected to an axon which in turn leads to multiple terminal buttons. Another type of neuron is the multipolar neuron which is similar to the bipolar neuron except that it has multiple dendritic trees attached to the soma. In all neurons, the length of the axon, the number of branches

that end in terminal buttons, and the number of dendritic spines varies.

Dendrites, soma, axons, and terminal buttons are all important components of how messages are passed from one neuron to other neurons. It starts when a neuron, called the presynaptic neuron, releases a chemical neurotransmitter into the synapse, a small space between a terminal button of the presynaptic neuron and a dendritic spine of the postsynaptic neuron, the neuron being studied. The neurotransmitter binds with spots on the dendritic spine of the postsynaptic neuron which creates a message via a current. Messages are passed down the tree to the soma where all messages are combined and integrated via the ion adjustment detailed above. If the summation of the incoming currents passes a threshold, an action potential occurs. The resulting voltage spike travels down the axon to the terminal buttons making them release their own chemical neurotransmitters. How the neurotransmitter is released depends on the type of neurotransmitter, but it will generally use channels that can be opened and closed. When the action potential reaches the terminal button a chain reaction occurs that ultimately opens up the channel, allowing the neurotransmitters to leave the terminal button and go into the synapse. Once in the synapse, the whole reaction starts again with the next neuron(s) who have dendritic spines on the other side of those synapses. The number of dendrites and their length to the soma can affect when they are integrated by the soma.

The brain has other supporting cells which can affect



how messages are formed and passed. These supporting cells, glia, are different types of neurons than those discussed above and include astrocytes, oligodendrocyte, and microglia. Astrocytes provide nutrients and regulate the extracellular fluid. Oligodendrocytes form the myelin sheaths which surround and insulate axons of the neurons allowing the action potential to travel down the axon better. Microglia protects the brain from invading microorganisms. The functions of these glia cells affect how efficiently neurons work.

Models break down neuron functions and properties by using mathematical equations. Models that are biologically accurate tend to use more equations than models that are less biologically accurate due to the fact that multiple biological components are explicitly being modelled to simulate the neuron's functioning.

Hodgkin-Huxley Model

The Hodgkin-Huxley model, one of the simplest biological models (Abbott & Kepler, 1990), uses four differential equations to compute the membrane potential. These four differential equations model the ionic flow of the neuron.

In the early 1950's, Hodgkin and Huxley studied the giant axon of a squid and used their findings to develop their model of the neuron (Hodgkin & Huxley, 1952b, 1952c, 1952d; Hodgkin & Keynes, 1955; Hodgkin & Huxley, 1952a; Hodgkin, Huxley, & Katz, 1952). They studied ionic flow by inserting an electrode into the cell and inputting a current to measure how the flow of ions and cell membrane change based on this inputted current. From their measurements, the researchers were able to derive detailed equations to explain the changes to the ionic current density. In general, they found that:

$$I = C_M \frac{dV}{dt} + I_i \quad (1)$$

where I is the total membrane current density measured in microamps per centimeter squared, C_M is the membrane capacity measured in microfarads per centimeter squared¹ which is assumed to be constant and equal to $1 \mu\text{F}/\text{cm}^2$, $\frac{dV}{dt}$ is the change in the displacement of the membrane potential from its resting value with respect to time and I_i represents the three different ionic current densities measured (Na^+ , K^+ , and leakage).

You will commonly see Eq. 1 written as:

$$C_M \frac{dV}{dt} = I - \sum_{ion} I_{ion} \quad (2)$$

where the summation is over all ionic currents measured. This equation is useful because it focuses on the change of the membrane potential.

The ionic current density is divided into the three ionic currents that Hodgkin and Huxley measured. These ions are sodium (Na^+), potassium (K^+) and a catch-all group called leakage. Leakage consists mostly of chloride but may also have small amounts of other ions. Despite large amounts of organic ions in the intracellular fluid, organic ions is not modelled because they don't move between the intra- and extracellular fluid. The three measured ionic currents are represented in equation 1 as I_i and in equation 2 as $\sum_{ion} I_{ion}$. Each of these ions has their own equation such that the summation:

$$\sum_{ion} I_{ion} = I_{\text{Na}^+} + I_{\text{K}^+} + I_{\text{Leak}} \quad (3)$$

is decomposed into:

$$I_{\text{Na}} = g_{\text{Na}^+} h m^3 (V - E_{\text{Na}^+}) \quad (4a)$$

$$I_{\text{K}} = g_{\text{K}^+} n^4 (V - E_{\text{K}^+}) \quad (4b)$$

$$I_{\text{Leak}} = g_{\text{Leak}} (V - E_{\text{Leak}}). \quad (4c)$$

Therefore:

$$\begin{aligned} \sum_{ion} I_{ion} &= g_{\text{Na}^+} h m^3 (V - E_{\text{Na}^+}) \\ &+ g_{\text{K}^+} n^4 (V - E_{\text{K}^+}) \\ &+ g_{\text{Leak}} (V - E_{\text{Leak}}). \end{aligned} \quad (5)$$

The parameters include constants g_i which are the maximum membrane conductances per ion measured in millisiemens per centimeter squared² and E_i are the value at which time there is no movement of the corresponding ion between the intra- and extracellular fluid (the reversal potential). The values of g_i and E_i were calculated by Hodgkin and Huxley to fit their empirical findings (see Table 1). The parameters h , m , and n are voltage-dependent conductance variables, also known as gating variables. Changes in these variables are calculated using differential equations (Equations 6a, 6b, and 6c). In all of the equations, i represents Na^+ , K^+ , and the leakage.

The gating variables represent the probability of the channels being open, of ions moving from intracellular to extracellular fluid, or vice versa. By definition the values of the gating variables (h , m , and n) can be anywhere between 0 and 1 and are dependent on both time and membrane voltage. The change in the gating variables is based

¹A farad is the capacitance in which one electric charge causes a potential difference of one volt.

²Siemens is the unit of electric conductance.

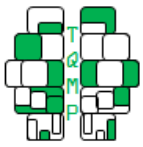


Table 1 ■ The constant parameters of the Hodgkin-Huxley model. In the original article, parameters are determined with the membrane resting potential (V_{rest}) set to 0 mV in other articles, a resting potential of -65 if used. To reflect either case, the above constants, and the Matlab code (Appendix A), add a parameter V_{rest} .

	E_i	g_i
Na ⁺	$115 + V_{rest}$ mV	120 mS/cm ²
K ⁺	$-12 + V_{rest}$ mV	36 mS/cm ²
Leak	$10.6 + V_{rest}$ mV	0.3 mS/cm ²

on the following differential equations:

$$\frac{dh}{dt} = \alpha_h(V)(1-h) - \beta_h(V)h \quad (6a)$$

$$\frac{dm}{dt} = \alpha_m(V)(1-m) - \beta_m(V)m \quad (6b)$$

$$\frac{dn}{dt} = \alpha_n(V)(1-n) - \beta_n(V)n \quad (6c)$$

The asymptotic value of any of the gating variables is:

$$\bar{x} = \frac{\alpha_x(V)}{\alpha_x(V) + \beta_x(V)} \quad (7)$$

where x stands for h , m , or n . You may see Equation 7 written as:

$$\bar{x} = \frac{\alpha_x(V)}{\tau_x(V)} \quad (8)$$

where τ is the time constant and equals $\alpha_x(V) + \beta_x(V)$. These equations are also called the Equilibrium functions. The above equations use α_i and β_i , which are formulas that Hodgkin and Huxley derived from their research (see Table 2).

There are two types of gating variables, activation (m and n) and inactivation (h) which represent the probability of the Na⁺ (h and m gates) and K⁺ (n gate) channels being open. Higher values, or probabilities, of activation gates means there is an increase in the depolarization of the cell membrane, while higher values of inactivation gates means there is a decrease in the depolarization of cell membrane. Figure 5 shows how the gate values change to stay in equilibrium dependent on the voltage. Na⁺ has the two forces working on it, which is represented by the h and n gates; with increased voltage is a decrease of inactivation, and an increase in activation of Na⁺ channels. K⁺ channels open later than Na⁺ channels, as seen by m raising after the n gate. The inactivation gate is the opposite of the activation gates, decreasing as the activation gates increase.

Table 2 ■ Equations used to define gating variables. V_{rest} is added to reflect that different resting potentials often used in varying articles (either 0 mV or -65 mV). See Appendix B for more information on the different forms of these equations.

	$\alpha_i(V)$	$\beta_i(V)$
h	$0.07e^{\frac{(V_{rest}-V)}{20}}$	$\frac{1}{1 + e^{3-0.1(V-V_{rest})}}$
m	$\frac{2.5 - 0.1(V - V_{rest})}{e^{2.5-0.1(V-V_{rest})} - 1}$	$4e^{\frac{(V_{rest}-V)}{18}}$
n	$\frac{0.1 - 0.01(V - V_{rest})}{e^{1-0.1(V-V_{rest})} - 1}$	$0.125e^{(V_{rest}-V)/80}$

Figure 6 shows the conductance of the ions in relation to an action potential. When an action potential starts, Na⁺ conductance drastically increases and at the peak of the action potential, Na⁺ conductance starts decreasing. This represents sodium's movement into and then out of the intracellular fluid by the opening and then closing of the Na⁺ channels. When an action potential starts, the sodium channels open, allowing the Na⁺ into the cell, but at the peak of the action potential the channels become refractory so Na⁺ can no longer enter but slowly leaves the cell by the forces working on the ions. The ion channels for K⁺ are slower to open and close which is also reflected in the conductance of K⁺.

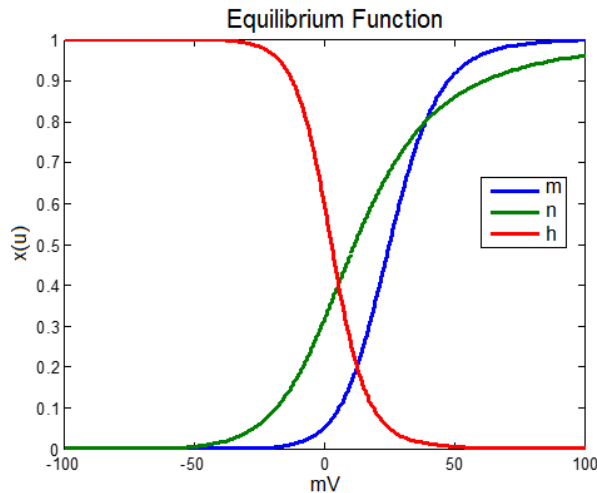
The four differential equations that make up the Hodgkin-Huxley model therefore are the three gating variables (Eqs 6a, 6b, and 6c) and the ionic current density equation (Eq 1). These equations are the mathematical equivalent of the ionic current: how movement of the ions affect the membrane potential and how the membrane potential affects the movement of the ions. Notice that the gating variables are affected by the voltage of the membrane and that the voltage is affected by the gating variables and ionic current. This is related to how the different ions can permeate the cell membrane depending on the membrane potential because different channels open and close depending on cell membrane voltage.

The parameter I in equations 1 and 2 is an input current, which means current coming in from an outside source. How the model handles an input current to produce action potentials is very important since it is the conversion of input current from other neurons into action potentials that is the basis of our brain's functionality.

Figure 7 displays some of the different neural responses that occur due to an input current. These responses are based on ionic current, therefore changes in the conductance and gating variables are also displayed. If the input is too low no action potential is produced (top



Figure 5 ■ Equilibrium Function of the gating variables (Eq. 11)



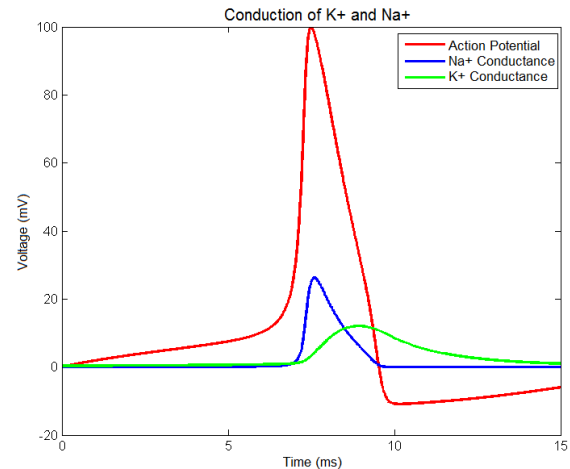
row). At the introduction of the input current the ions fluctuate until they reach a new steady state with slightly higher voltages. When a slightly higher voltage is injected into the node (middle row) a potential might be induced before stabilization occurs. In terms of biology, this is a redistribution of the ions due to the changes of electrostatic pressure. When the input current reaches approximately $6.3 \mu\text{A}/\text{cm}^2$ a steady stream of action potentials occurs (bottom row). Notice though, that there is an increasing amount of time between each spike in the regular spiking pattern; this increase in the interval between spikes is an important and basic component of spikes. The Hodgkin-Huxley model explains the membrane potential and thus action potentials via ionic current changes. Because the model is strongly based on biological principles, the results it produces are accurate in both timing and voltage, important characteristics in modelling.

Implementation of the Hodgkin-Huxley Model

The majority of equations used in modelling the Hodgkin-Huxley model are differential equations; these equations represent the change in the variable over time (change in voltage, change in gating), not the actual membrane potential or gating value at a given time. The functions to determine the value at a given time is the integral of the differential equation. The problem with that is that the four differential equations are interrelated and so it is hard to calculate the integral. Therefore an approximation of the function is used, often by Euler's method (see Appendix C for a review of Euler's method).

The Hodgkin-Huxley model is fitted to the squid's giant axon so the Hodgkin and Huxley did the work in parameter

Figure 6 ■ Conductance of ions in relation to an action potential. Total time illustrated is 15 ms with a steady input current of 2 mV



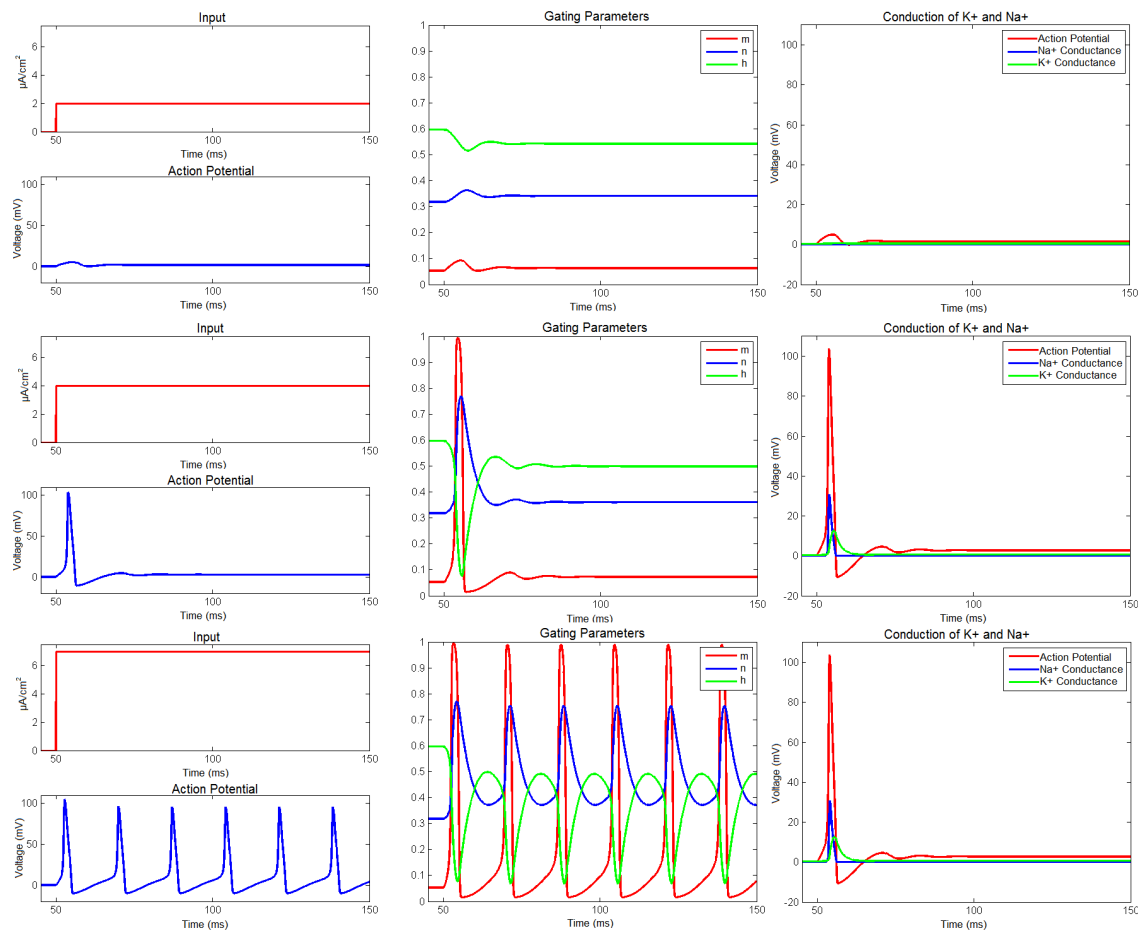
selection; they figured out the necessary values for the reversal potential, and membrane conductance as well as the full equations for the gating variables. Besides the input current, another parameter occurs because of the integral approximation method used. In Euler's method, there is a parameter that is the time step, dt , which depending on the value selected may make it impossible for the model to work, or make the model take too long to work. The time step is directly related to the time it takes to run, so the smaller the value, the longer simulations will take. All simulations used in this tutorial set dt to 0.01. For a rough approximation of time differences, when $dt=0.01$, a 150 ms simulation took roughly 250 ms to run on a basic laptop running Windows 7 and Matlab 2013. The same simulation at $dt=0.1$ took approximately 75 ms and $dt=0.001$ took approximately 1650 ms. While it may seem like using a larger dt makes the most sense, dt is also related to how accurate the simulation is. Having a large dt can mean some necessary steps in ionic flow are skipped such as the opening and closing of voltage dependent gates. Therefore the simulation becomes inaccurate. Specifically in the case of Hodgkin-Huxley, larger dt values can drastically affect the ionic current so that the spikes voltage is higher value than the code can handle, or is not realistically plausible, which will stop the code from functioning.

Conclusion

The Hodgkin-Huxley model is one the simplest models in the explicit representation of neurons category. Yet, it requires 4 differential equations plus 8 other equations and 3 parameters to model just the ionic flow of a neuron. Because of the complexity of the model, an approximation



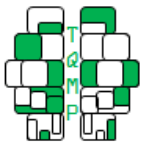
Figure 7 ■ Three examples of how an input current produces an action potential via ionic currents. The first 50ms had no input current to make sure the ionic flows stabilize from the initial values used. Input current starts at 50ms as shown. Rows vary by input current of $2 \mu\text{A}/\text{cm}^2$, $4 \mu\text{A}/\text{cm}^2$, and $7 \mu\text{A}/\text{cm}^2$ for top, middle, and bottom rows respectively. The first column shows cell membrane effect based on input, second column is changes on the gating variables, while third column is changes in conduction of K^+ and Na^+ with the action potential



method is needed to calculate the integrals of the differential equations. This approximation adds another parameter that needs to be carefully reviewed since that parameter affects how long the simulation takes to run and how accurate the simulation is.

References

- Abbott, L. F. & Kepler, T. B. (1990). Model neurons: from hodgkin-huxley to hopfield. In T. Hanks (Ed.), *Statistical mechanics of neural networks* (pp. 5–18). Berlin: Springer. doi:[10.1007/3540532676_37](https://doi.org/10.1007/3540532676_37)
- Carlson, N. R. & Birkett, M. A. (2016). *Physiology of behavior* (12 edition). Boston: Pearson.
- DasGupta, B. & Schnitger, G. (1994). The power of approximating: a comparison of activation functions. *MATHEMATICAL RESEARCH*, 79, 641–641. doi:[10.1.1.52.263](https://doi.org/10.1.1.52.263)
- Denker, J., Schwartz, D., Wittner, B., Solla, S., Howard, R., Jackel, L., & Hopfield, J. (1987). Large automatic learning, rule extraction, and generalization. *Complex Systems*, 1(5), 877–922.
- Dreiseitl, S. & Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. *Journal of Biomedical Informatics*, 35(5-6), 352–359. doi:[10.1016/S1532-0464\(03\)00034-0](https://doi.org/10.1016/S1532-0464(03)00034-0)



- Dubois, M. L. (2010). *Action potential: biophysical and cellular context, initiation, phases and propagation (1 edition)*. New York: Nova Science Publishers, Inc.
- Gerstner, W. (2002). *Spiking neuron models: single neurons, populations, plasticity*. Cambridge, UK; New York: Cambridge University Press.
- Hodgkin, A. L. & Huxley, A. F. (1952a). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology*, 117(4), 500–544. doi:10.1113/jphysiol.1952.sp004764
- Hodgkin, A. L. & Huxley, A. F. (1952b). Currents carried by sodium and potassium ions through the membrane of the giant axon of loligo. *The Journal of Physiology*, 116(4-4), 449–472. doi:10.1113/jphysiol.1952.sp004717
- Hodgkin, A. L. & Huxley, A. F. (1952c). Movement of sodium and potassium ions during nervous activity. (pp. 43–52). Washington: Cold Spring Harbor Laboratory Press. doi:10.1101/sqb.1952.017.01.007
- Hodgkin, A. L. & Huxley, A. F. (1952d). Propagation of electrical signals along giant nerve fibres. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 177–183. doi:10.1098/rspb.1952.0054
- Hodgkin, A. L., Huxley, A. F., & Katz, B. (1952). Measurement of current-voltage relations in the membrane of the giant axon of loligo. *The Journal of Physiology*, 116(4), 424. doi:10.1113/jphysiol.1952.sp004716
- Hodgkin, A. L. & Keynes, R. D. (1955). Active transport of cations in giant axons from sepia and loligo. *The Journal of Physiology*, 128(1), 28–60. doi:10.1113/jphysiol.1955.sp005290
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. 79, 2554–2558. doi:10.1073/pnas.79.8.2554
- Kuebler, E. S. & Thivierge, J.-p. (2014). Asynchronous coding in neuronal networks. *BMC Neuroscience*, 15(1), 1–2. doi:10.1186/1471-2202-15-S1-P26
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. doi:10.1038/nature14539
- Maass, W. (1997). Networks of spiking neurons: the third generation of neural network models. *Neural Networks*, 10(9), 1659–1671. doi:10.1016/s0893-6080(97)00011-7
- McCulloch, W. S. & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115–133. doi:10.1007/bf02478259
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408. doi:10.1037/h0042519
- Shen, Y. & Bax, A. (2013). Protein backbone and sidechain torsion angles predicted from nmr chemical shifts using artificial neural networks. *Journal of Biomolecular NMR*, 56(3), 227–241. doi:10.1007/s10858-013-9741-y
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In M. Welling, C. Cortes, N. D. Lawrence, & K. Q. Weinberger (Eds.), *Z. ghahramani* (pp. 3104–3112). Inc: Advances in Neural Information Processing Systems 27. Curran Associates.
- Taigman, Y., Yang, M., Ranzato, M., & Wolf, L. (2014). Deepface: closing the gap to human-level performance in face verification. In IEEE (Ed.), *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1701–1708). Washington: IEEE. doi:10.1109/cvpr.2014.220
- Zamani, M., Sadeghian, A., & Chartier, S. (2010). A bidirectional associative memory based on cortical spiking neurons using temporal coding. In T. Hanks (Ed.), *The 2010 international joint conference on neural networks* (pp. 1–8). Washington: Institute of Electrical and Electronics Engineers. doi:10.1109/IJCNN.2010.5596806

Appendix A: Matlab code for Hodgkin-Huxley

Helper Functions

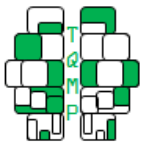
The gating variables use equations α and β (see Table 2) are separated into their own functions. These functions take as input the current voltage and the resting membrane voltage and output the gating value. In Matlab, these functions need to be in either their own files or at the bottom of the main file.

% calculate alpha m and beta m based on Table 2

```
function [alpha_m, beta_m] = m_equations(V, Vrest)
    alpha_m = (2.5-0.1*(V-Vrest)) / (exp(2.5-0.1*(V-Vrest))-1);
    beta_m = 4*exp((Vrest-V)/18);
end
```

% calculate alpha n and beta n based on Table 2

```
function [alpha_n, beta_n] = n_equations(V, Vrest)
    alpha_n = (0.1-0.01*(V-Vrest)) / (exp(1-0.1*(V-Vrest))-1);
```

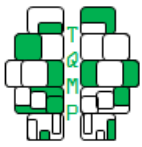


```
beta_n = 0.125*exp((Vrest-V)/80);  
end  
% calculate alpha h and beta h based on Table 2  
function [alpha_h, beta_h] = h_equations(V, Vrest)  
    alpha_h = 0.07*exp((Vrest-V)/20);  
    beta_h = 1/(1+exp(3-0.1*(V-Vrest)));  
end
```

Main Function

Below is the main Hodgkin-Huxley function. The code integrates an input current via the differential equations to produce a vector of membrane voltage values over the time period. It is set up to allow the user to adjust the resting potential, V_{rest} , to any value they want; in Hodgkin-Huxley's papers, this is set to 0 mV but in many other papers, this is set to -65 mV. The time step, dt , is set to 0.01 which is the value used in all the graphs in this paper. The duration of the simulation can be adjusted based on the needs of the research through the variable `totalTime`. Initial values for the ionic currents are their equilibrium values and the initial values of voltage is the resting potential.

```
function HodgkinHuxley  
    Vrest = 0; % mV – change this to –65 if desired  
    dt = 0.01; % ms  
    totalTime = 150; % ms  
    C = 1; % uF/cm^2  
  
    % constants; values based on Table 1  
    E_Na = 115 + Vrest; % mV  
    E_K = -6 + Vrest; %mV  
    E_Leak = 10.6 + Vrest; % mV  
  
    g_Na = 120; % mS/cm^2  
    g_K = 36; % mS/cm^2  
    g_Leak = 0.3; % mS/cm^2  
  
    % Vector of timesteps  
    t = [0:dt:totalTime];  
  
    % Current input – change this to see how different inputs affect the neuron  
    I_current = ones(1, length(t))*0.0;  
    I_current(50/dt:end) = 3; % Input of 3 microA/cm2 beginning at 50 ms and steady until end of time period.  
  
    % initializing values  
    V(1) = Vrest; % membrane potential is starting at its resting state  
  
    % separate functions to get the alpha and beta values  
    [alphaM, betaM] = m_equations(V(1), Vrest);  
    [alphaN, betaN] = n_equations(V(1), Vrest);  
    [alphaH, betaH] = h_equations(V(1), Vrest);  
  
    % initializing gating variables to the asymptotic values when membrane potential  
    % is set to the membrane resting value based on equation 13  
    m(1) = (alphaM / (alphaM + betaM));  
    n(1) = (alphaN / (alphaN + betaN));  
    h(1) = (alphaH / (alphaH + betaH));
```



```
% repeat for time determined in totalTime, by each dt
for i = 1:length(t)
    % calculate new alpha and beta based on last known membrane potential
    [alphaN, betaN] = n_equations(V(i), Vrest);
    [alphaM, betaM] = m_equations(V(i), Vrest);
    [alphaH, betaH] = h_equations(V(i), Vrest);

    % conductance variables – computed separately to show how this
    % changes with membrane potential in one of the graphs
    conductance_K(i) = g_K*(n(i)^4);
    conductance_Na(i) = g_Na*(m(i)^3)*h(i);

    % retrieving ionic currents
    I_Na(i) = conductance_Na(i)*(V(i)-E_Na);
    I_K(i) = conductance_K(i)*(V(i)-E_K);
    I_Leak(i) = g_Leak*(V(i)-E_Leak);

    % Calculating the input
    Input = I_current(i) - (I_Na(i) + I_K(i) + I_Leak(i));

    % Calculating the new membrane potential
    V(i+1) = V(i) + Input*dt*(1/C);

    % getting new values for the gating variables
    m(i+1) = m(i) + (alphaM*(1-m(i)) - betaM*m(i))*dt;
    n(i+1) = n(i) + (alphaN*(1-n(i)) - betaN*n(i))*dt;
    h(i+1) = h(i) + (alphaH*(1-h(i)) - betaH*h(i))*dt;

end
end
```

Graphs

All the graphs used in this paper were made using the code presented here. The code for the graphs should be in the main function code before the last end. First graph is the Gating Parameters graph used in the middle column of Figure 7. Input is set to begin at 50 ms, and the graph ignores the first 45 ms because this time is to allow for the voltage to stabilize around the initial values.

```
figure('Name', 'Gating Parameters')
plot(t(45/dt:end), m(45/dt:end-1), 'r', t(45/dt:end), n(45/dt:end-1), 'b', t(45/dt:end), h(45/dt:end-1), 'g', 'LineWidth', 2)
legend('m', 'n', 'h')
xlabel('Time (ms)')
ylabel('')
title('Gating Parameters')
```

The second graph is the double graph of input (top) and voltage (bottom) used in Figure 7, first column. As with above, time starts at 45 ms in, shortly before input is injected into the neuron.

```
figure('Name', 'Membrane Potential vs input')
subplot(2,1,1)-
plot(t(45/dt:end), V(45/dt:end-1), 'LineWidth', 2)
xlabel('Time (ms)')
```




```
ylabel('Voltage (mV)')
title('Action Potential')
subplot(2,1,2)
plot(t(45/dt:end), I_current(45/dt:end), 'r', 'LineWidth', 2)
xlabel('Time (ms)')
ylabel('Voltage (mV)')
title('Input')
```

Figures 6 and 7 (last column) use the below graph to show conductance in relation to an actual potential.

```
figure('Name', 'Conductance')
plot(t(45/dt:end), V(45/dt:end-1), 'r', t(45/dt:end), conductance_Na(45/dt:end), 'b',
      t(45/dt:end), conductance_K(45/dt:end), 'g', 'LineWidth', 2)
legend('Action Potential', '\ch{Na+} Conductance', '\ch{K+} Conductance')
xlabel('Time (ms)')
ylabel('Voltage (mV)')
title('Conduction of \ch{K+} and \ch{Na+}')
```

Equilibrium (Figure 5) graph shows values needed for the grating variables to hold equilibrium over a range of voltage values. This graph does not need the Hodgkin-Huxley function, but does need the helper functions to run. The code starts by calculating all the equilibrium values for the voltage range (-100 mV to 100 mV) before the actually graph is created.

% Special graph to show ionic current movement

```
Vrest = 0;
voltage = [-100:0.01:100];
for i = 1:length(voltage)
    [alphaN, betaN] = n_equations(voltage(i), Vrest);
    [alphaM, betaM] = m_equations(voltage(i), Vrest);
    [alphaH, betaH] = h_equations(voltage(i), Vrest);
    taum(i) = 1/(alphaM+betaM);
    taun(i) = 1/(alphaN+betaN);
    tauh(i) = 1/(alphaH+betaH);
    xm(i) = alphaM/(alphaM+betaM);
    xn(i) = alphaN/(alphaN+betaN);
    xh(i) = alphaH/(alphaH+betaH);

    aN(i) = alphaN;
    bN(i) = betaN;

    aM(i) = alphaM;
    bM(i) = betaM;

    aH(i) = alphaH;
    bH(i) = betaH;
end

figure('Name', 'Equilibrium Function');
plot(voltage, xm, voltage, xn, voltage, xh, 'LineWidth', 2);
legend('m', 'n', 'h');
title('Equilibrium Function');
xlabel('mV');
ylabel('x(u)');
```



Table 3 ■ Different forms that $\alpha_i(V)$ and $\beta_i(V)$ can take (Table 2). In the first set, $V = V_{resting} - V_{current}$. The second set of equations has the resting potential is set to 0 mV and u is the membrane potential. The final set has the resting potential as -65 mV and V is the membrane potential .

	$\alpha_i(V)$	$\beta_i(V)$
h	$0.07e^{\frac{V}{20}}$	$\frac{1}{e^{\frac{V+30}{10}} + 1}$
m	$\frac{0.1(V+25)}{e^{\frac{V+25}{10}} - 1}$	$4e^{\frac{V}{18}}$
n	$\frac{0.01(V+10)}{e^{\frac{V+10}{10}} - 1}$	$0.125e^{V/80}$
h	$0.07e^{\frac{-u}{20}}$	$\frac{1}{e^{3-0.1u} + 1}$
m	$\frac{2.5 - 0.1u}{e^{2.5-0.1u} - 1}$	$4e^{\frac{-u}{18}}$
n	$\frac{0.1 - 0.01u}{e^{1-0.1u} - 1}$	$0.125e^{-u/80}$
h	$0.07e^{-0.5(V+65)}$	$\frac{1}{1 + e^{-0.1(V+35)}}$
m	$\frac{0.1(V+40)}{1 - e^{-0.1(V+40)}}$	$4e^{-0.0556(V+65)}$
n	$\frac{0.01(V+55)}{1 - e^{-0.1(V+55)}}$	$0.125e^{-0.0125(V+65)}$

Appendix B: Ionic Current Equations

When researching Hodgkin-Huxley, equations in Table 2 appeared in a number of different forms. These different forms occur because they use different resting membrane potential.

The actually formulas given by Hodgkin and Huxley (Hodgkin & Huxley, 1952a) are seen in the top part of Table 3 in which V is the change of the membrane from its resting potential; $V = V_{resting} - V_{current}$. The formulas used in Table 3 are the same as those above, except in the code above V is explicitly replaced with variables for the resting membrane potential and the current membrane potential.

When the resting membrane potential is 0 mV, the variable $V_{resting}$ is replaced with 0, and the formula is as shown in the middle of Table 3.

While Hodgkin and Huxley adjusted all their constants so that the resting potential is 0 mV, in reality the resting potential is around -65 mV for the giant squid axon. Therefore, the equations often shown are those in the bottom of Table 3 (Abbott & Kepler, 1990).

All the above equations show the same information but in slightly different ways. Because of the overlap in variables used and difference in variable meanings, it is important to carefully read exactly what the authors are modelling.

Depending on what the resting potential is, the reversal potential (E_i) also needs to be adjusted as shown in Table 1.

Appendix C: Euler Method

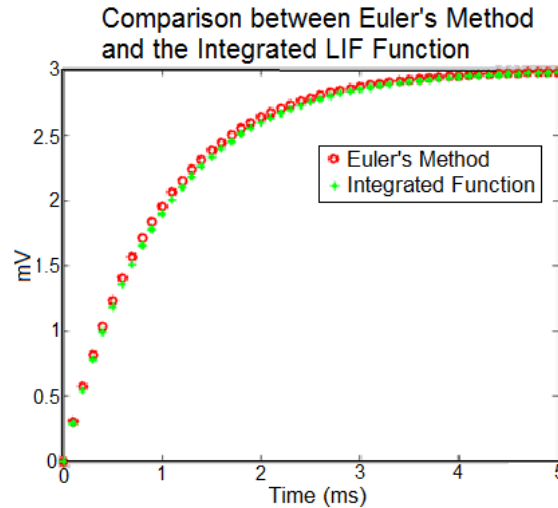
The Euler method is a quick and easy method used to estimate a function from its ordinary differential equation (ODE). It is a step-wise calculation based on the idea that the tangents, if close enough, can provide an estimate of the unknown function. In simple terms to solve a differential equation with the Euler method:

1. Calculate starting values based on prior knowledge
2. Calculate the differential equation, using values obtained in step 1 (or step 4 if repetition)
3. Multiple results from step 2 by a small time step, dt (often 0.1 or 0.01)
4. Add values obtained from step 3 to the starting values used in step 2
5. Repeat steps 2 to 4

As an example, let's look at the leaky integrate and fire (LIF) model (see part 2 of this tutorial). The LIF model was chosen because it is possible to have both the differential equation and the integrated function therefore the results of the Euler method can be compared with the actual function. The differential equation for the LIF is: $\tau \frac{dv(t)}{dt} = RI(t) - v(t)$



Figure 8 ■ Comparison of results using the integrated function (last firing at time 0) and Euler's method for 5 ms with time step 0.1



which can be rewritten as: $\frac{dv(t)}{dt} = \frac{RI(t)-v(t)}{\tau}$ The first step is to pick initial values for the functions. For convenience, let $v(0) = 0$ (starting membrane potential), $R = 1$ and $\tau = 1$. Let's assume that there is a constant incoming current of $3 \mu\text{A}/\text{cm}^2$ so that $I(t) = 3$ for all values of t . Let the time step, dt , equal 0.1. Therefore, following the steps:

Step 1: $v(0) = 0$

Step 2: $\frac{RI(t)-v(t)}{\tau} = \frac{1(3)-0}{1} = 3$

Step 3: $3dt = 3 \times 0.1 = 0.3$

Step 4: $v(1) = v(0) + \text{step 3} = 0 + 0.3 = 0.3$

Step 5 is the repetition of the previous steps. Therefore, $v(2) = 0.57, v(3) = 0.813, v(4) = 1.0317, \dots$

The steps can be written into one mathematic equation

$$v(t) = v(t-1) + \left(\frac{RI(t) - v(t-1)}{\tau} \right) dt \quad (9)$$

Let

$$F(t) = \frac{RI(t) - v(t-1)}{\tau} \quad (10)$$

to obtain the generalized Euler's method used in the code for the other neuron models.

$$v(t) = v(t) + F(t) dt \quad (11)$$

While the Euler method is not an exact solution to the differential equation, but an estimate, it does a good job provided that the time step, dt , is small enough.

Citation

Johnson, M. G. & Chartier, S. (2017). Spike neural models (part I): The Hodgkin-Huxley model. *The Quantitative Methods for Psychology*, 13(2), 105–119. doi:10.20982/tqmp.13.2.p105

Copyright © 2017, Johnson and Chartier. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Received: 13/09/2016 ~ Accepted: 19/01/2017