'A simulation study of spatio-temporal interactions in a coincidence detection neuron'

Maartje van der Veen
August 2001
University of Groningen

Supervisors:
Tjeerd Andringa
Henk Mastebroek
Erwin Mulder

Contents

Preface

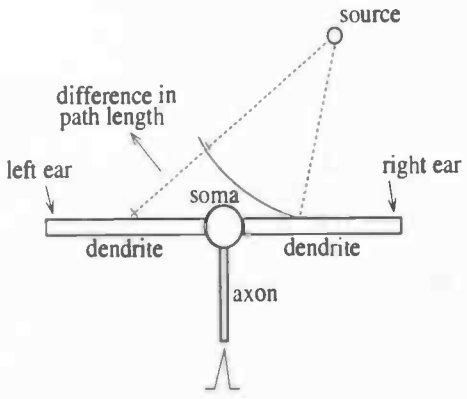

Figure 1: A coincidence detector neuron

In the auditory system of mammals and birds, sounds are localized with coincidence detection neurons. These neurons are bipolar, with each dendrite receiving input from only one ear. The beginning of the sound arrives at different times at the dendrites. The neuron compares the time difference in this arrival to localize it. This is shown in Figure 1.
In Nature [vol 393, 1998, page 268-272], Agmon-Snir et al. showed that dendrites of coincidence detector cells in the auditory brain stem help to improve sound localization.
The results they discuss reveal that they used an unrealistically short dendrite in their simulation, which is anatomically incorrect.
In nature, dendrites are longer and this must have a function, otherwise, these long dendrites would not have prevailed in the auditory system until today. The question rises why the dendrites are longer in reality.

A neuron with longer dendrites is more likely stimulated at more than one location (Figure 2). Stimulations at different locations might give rise to a sensitivity to specific spatio-temporal properties other than the onset of the sound. One could think of frequency selectiveness for certain tones which are comprised of a base frequency and one (or more) harmonics. The main question in this study will be gear towards these suspicions: can a longer dendrite improve the frequency sensitivity, to better ascertain the frequency and direction of the provided signal?

To test this conjecture, I will design a coincidence detection neuron in the "GEneral NEural SImulation System" (GENESIS), with parameter settings

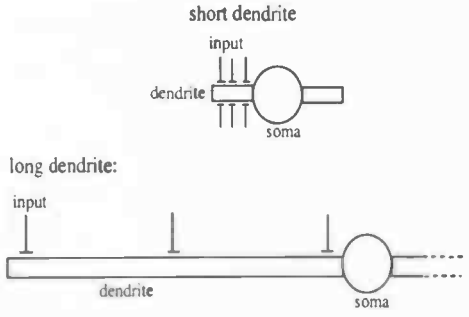

Figure 2: Difference in using a short or a long dendrite.

within ranges known from the literature.

The structure of this report is as follows:
Section 1 studies the behaviour of neurons in more detail, with topics as the membrane equation, an RC-circuit, the cable equation and the characteristic properties of a neuron.
Section 2 deals with the design of the model in the simulation program GENESIS. The bipolar neuron is designed with objects and messages.
In Section 3, the simulations are performed.
Section 3.1 is a study for the robustness of the model for variations in dendritic time constant τ and space constant λ and for choosing realistic physiological parameter values.
Section 3.2 gives some basic simulations as a validation of the model.
Next to these starting simulations, the real simulations are performed and evaluated in Sections 3.3 to 3.5. Section 3.3 deals with the design of an optimal response by providing two inputs with a particular interstimulus time. In Section 3.4, these inputs are used to design a pulse train. Section 3.5 gives the results of providing a frequency higher or lower than the frequency of the traveling time of the dendrite and what happens if a harmonic is simulated.
Section 4 summarizes the conclusions.

Chapter 1

The Neuron

1.1 The Membrane Equation

Each cell has an electric equilibrium potential due to the diffusion of ions. This *membrane potential* is defined as the difference between the inner and outer potential of the cell,

$$V_m(t) = V_i(t) - V_o(t). \tag{1.1}$$

Much of the behaviour of the cell can be explained with Ohm's law:

$$\Delta V = IR_m. \tag{1.2}$$

For the membrane, I is the diffusion current and R_m the membrane resistance.

In rest, the potential is negative. The resting potential is actually a dynamic equilibrium. This means that although the potential is at a stable value, there are still currents going through the membrane. The flow of ions out of the cell equals the flow into the cell, because the electric force is countered by the diffusion velocity. The diffusion velocity depends on the concentration of ions in and outside the cell. The electric force depends on the potential difference which is caused by the replacing of charge by diffusion. It costs energy to keep the cell in this desired active equilibrium position.
Currents around the membrane consist of two major components: one charges the membrane capacitance and the other is associated with the specific types of ions across the membrane.

The diffusion current through the membrane gives rise to a potential difference. The membrane behaves like a capacitance, where the inside and

outside of the membrane together have a particular capacitance C_m with charge Q. The potential is defined as:

$$V = \frac{Q}{C_m}. \tag{1.3}$$

Differentiating this equation leads to an equation of the rate of change of the potential:

$$\frac{\partial V}{\partial t} = \frac{I}{C_m}. \tag{1.4}$$

1.2 Injection Current

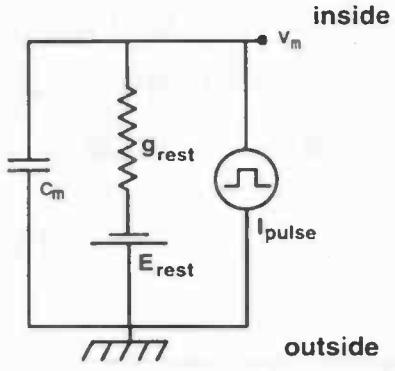

Figure 1.1: Equivalent circuit for electrical model of an isopotential nerve membrane. Only the passive membrane elements are shown.

When a current I_{inj} is injected in the neuron, only the passive channels influence the current. The total conductance of all transmembrane ion channels, opened during the rest situation E_{rest}, is termed g_{rest}. This is the input conductance with value that is reciprocal with the input resistance. The membrane is represented as a simple RC circuit, with I_{inj} parallel to C_m and g_{rest} in series with R_{rest}, as shown in Figure 1.1. V_m is the voltage difference between the inside and the outside of the cell. I_c is the capacitive current which charges the membrane capacitance.
The cell membrane can be modelled by many small RC circuits. If the dimensions of the cell are very small, the electric potential across the membrane is the same everywhere. Any spatial dependencies can be neglected. The cell is *isopotential*. This means that the behaviour of the cell can be described by a single RC component with a current source.

Charge must go somewhere; it never disappears. Therefore, the sum of all charges must be constant and the sum of all currents must therefore be equal to 0.
The sum of the internal currents must be equal to the injected current:

$$I_c + I_{rest} = I_{inj}. \tag{1.5}$$

This equals:

$$C_m \frac{dV_m(t)}{dt} + g_{rest}(V_m - E_{rest}) = I_{inj}. \tag{1.6}$$

E_{rest} is set to 0.
With an injection pulse with $V_m(0) = 0$ starting at t=0, the voltage changes according to:

$$V_m(t) = I_{inj} R_m (1 - e^{-t/\tau_m}), \tag{1.7}$$

where $\tau_m = C_m / g_{rest} = R_m Cm$. When $t = \tau_m$ is

$$V_m(t) = I_{inj} R_m (1 - e^{-1}) = 0.63 I_{inj} R_m. \tag{1.8}$$

1.3 Synaptic Input

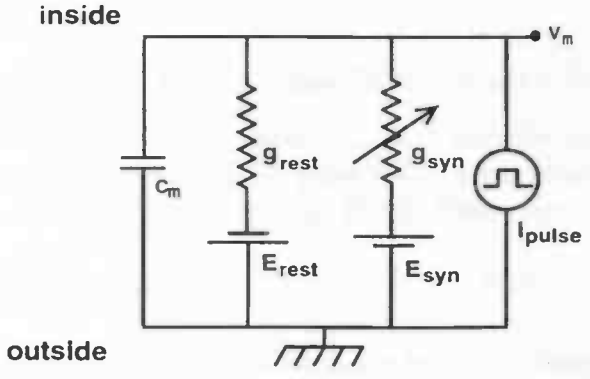

Figure 1.2: Equivalent cicuit for electrical model of an isopotential nerve membrane, with an additional conductive branch added parallel with the passive elements to represent the synaptic channels in the membrane.

The input of a synapse leads to a local conductance change of the membrane at the place of input. The opening of synaptic channels in an isopotential patch of membrane is modeled by a time-dependent conductance change

($g_{syn}(t)$. This conductance lies in series with a battery E_{syn}, the synaptic reversal potential, shown in Figure 1.2. I_{syn} is the synaptic current which flows through the channels. This current equals, according to Ohm's current law:

$$I_{syn}(t) = g_{syn}(t)(V_m - E_{syn}). \tag{1.9}$$

When g_{syn} increases because of more open channels, the difference between V_m and E_{syn} will always reduce, causing the cell to reduce to an equilibrium state. There are three situations:

$E_{syn} > V_m$: the negative synaptic current flows inside and depolarizes the cell, making V_m more positive.

$V_m > E_{syn}$: the positive synaptic current flows outside and hyperpolarizes the cell, making V_m more negative.

$V_m = E_{syn}$: there is no synaptic current because $(V_m - E_{syn}) = 0$ and, as a consequence no change in V_m.

The equation for synaptic current is:

$$I_c + I_{rest} + I_{syn} = 0, \tag{1.10}$$

or

$$C\frac{dV_m(t)}{dt} + g_{rest}V_m + g_{syn}(t)(V_m - E_{syn}) = 0. \tag{1.11}$$

When the synaptic channels are open ($g_{syn} > 0$) and the synaptic conductance change is seen as a rectangular pulse with an amplitude of g_{syn} and a duration of t_{syn}, the solution is (see also [Koch, 1999, page 10]):

$$V_m(t) = \frac{g_{syn}}{g_{syn} + g_{rest}}E_{syn}(1 - e^{-t(g_{syn}+g_{rest})/C_m}), \quad \text{for } 0 \leq t \leq t_{syn} \tag{1.12}$$

The steady-state solution, obtained when the synaptic channels are opened for an infinitely long duration ($t \to \infty$) is:

$$V_m = \frac{g_{syn}}{g_{syn} + g_{rest}}E_{syn} = \frac{1}{1 + g_{rest}/g_{syn}}E_{syn}. \tag{1.13}$$

The Equations (1.12) and (1.13) are strictly applicable only to step-conductance changes. Nevertheless, they provide general insights into the functional consequences of synaptic mechanisms. One is that, unless $E_{syn} = 0$, the membrane potential is always smaller than E_{syn}. Secondly, one can

see that V_m is a nonlinear function of g_{syn}. For example, Equation (1.13) tells us that if $g_{syn} = g_{rest}$, then $V_m = \frac{1}{1+1}E_{syn} = E_{syn}/2$. Assuming that $E_{syn} = 90\ mV$, the steady-state value of the potential is 45 mV. With a multiplication of g_{syn} by a factor of two, i.e. $g_{syn} = 2g_{rest}$, the steady depolarization is not 90 mV, as expected in the linear case, but 60 mV. Synaptic inputs sum nonlinearly, because the driving force for excitatory synaptic currents decreases with depolarization.
This nonlinearity implies that successive synaptic inputs will not sum linearly with each other.

1.3.1 Several Synapses

More than one synapse can be active at the same time. The resulting equation of this circuit is similar to Equation (1.11):

$$C_m\frac{dV_m}{dt} + g_{rest}V_m + g^1_{syn}(t)(V_m - E^1_{syn}) + g^2_{syn}(t)(V_m - E^2_{syn}) + \ldots = 0. \quad (1.14)$$

Each synaptic input may have a different reversal potential E_{syn} and a different corresponding conductance change g_{syn}, which may be activated at different times. The inputs can be hyperpolarizing or depolarizing, depending on the sign of $(V_m - E_{syn})$.
The solution is the nonlinear sum of the effects of all the synaptic inputs,

$$V_m(t) = \frac{g^1_{syn}E^1_{syn} + g^2_{syn}E^2_{syn} + \cdots}{g_{total}}(1 - e^{-g_{total}t/C_m}), \quad (1.15)$$

where

$$g_{total} = g_{rest} + g^1_{syn} + g^2_{syn} + \ldots \quad (1.16)$$

1.3.2 Alpha Function

A synaptic conductance change is better described by a smooth function, instead of a rectangular pulse. A good approximation for the smooth shape of the experimentally observed synaptic conductance change is the *alpha function*,

$$g_{syn}(t) = g_{max}\frac{t}{t_p}e^{1-t/t_p}. \quad (1.17)$$

This function increases rapidly to a maximum of g_{max} at $t = t_p$ and, after reaching that maximum, decreases more slowly to zero.

The GENESIS simulator uses a more general form, the dual exponential function,

$$g_{syn}(t) = \frac{Ag_{max}}{\tau_1 - \tau_2}(e^{-t/\tau_1} - e^{-t/\tau_2}), \text{for } \tau_1 > \tau_2. \quad (1.18)$$

A is a normalization constant chosen so g_{syn} reaches a maximum value of g_{max}. When $\tau_1 = \tau_2 = t_p$, this is equivalent to the alpha function.

1.4 The Hodgkin-Huxley Model

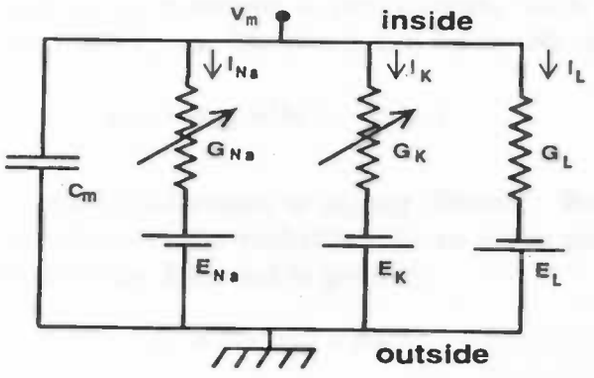

Figure 1.3: Electrical equivalent circuit proposed by Hodgkin and Huxley for a short segment of an axon

Hodgkin and Huxley developed a similar model with an equivalent circuit of the form shown in Figure 1.3. Here the current that moves the specific types of ions across the membrane is subdivided into different components for different ions, namely a sodium current I_{Na}, a potassium current I_K and a leakage current I_L, primarily for chloride ions. Each current has its own conductance G, the reciprocal of resistance, $G=1/R$, and its own equilibrium potential. The current is proportional to the multiplication of the conductance and the driving force. The net current is, again, 0. In this HH-model, the total ion current for the different ions is

$$I_{ion} = G_{Na}(V_m - E_{Na}) + G_K(V_m - E_K) + G_L(V_m - E_L). \quad (1.19)$$

The differential equation for this electrical circuit is described as:

$$C_m \frac{dV_m}{dt} + I_{syn} = I_{inj}. \quad (1.20)$$

1.5 Gate Processes

When a current flows, the ions pass through the ion channels. Different gates arrange the conductance of the ion flow in an individual ion channel. For sodium, two opposite gate processes provide for ion passing, activation and inactivation. This actually means that there are two types of gates, named m and h. In one channel, three m gates control the activation and one h the inactivation. The value for m equals the probability that the gate is open. This value depends on the membrane potential. The probability for all the gates to be in permeable state is m^3h. Only when all gates in a channel are opened, an ion is allowed to pass through. With only a single gate closed, the ion cannot pass. The description for the Na conductance is given by

$$I_{Na} = \overline{g}_{Na} m^3 h (V_m - E_{Na}). \tag{1.21}$$

The potassium channel conductance is slightly different. Four activation gates n control the channel. The probability for an ion to pass through is n^4. The description for the K current is given by:

$$I_K = G_K (V_m - E_K), \tag{1.22}$$

where

$$G_K = \overline{g}_K n^4. \tag{1.23}$$

The conductance is a measure for the number of open pores. One individual open channel contributes only marginally to the conductance. The higher the conductance, the more open channels. The maximum conductance, when all channels are open, is termed $\overline{g}$.

1.6 The Action Potential

An action potential is initiated when the membrane potential exceeds the particular threshold value for firing. Above that value the membrane potential increases very rapidly. An action potential is an all-or-none mechanism. The voltage-sensitive ion channels remain closed until the potential on the axon hillock has reached the threshold through depolarization, between 10 and 20 mV (when the resting potential is 0). In rest, the ion permeability of a cell membrane is low ($V = V_{rest}$). During activity, when $V \neq V_{rest}$, this permeability is high.
An injection current depolarizes the cell and Na channels are opened. These

ions begin to neutralize the internal negative ion concentration and further raise the internal potential towards the reversal potential of Na. In a run-away cycle, this will further increase the potential. The net effect is that the membrane depolarizes very rapidly: the membrane generates an action potential. The values of the activation and inactivation constants depend on the membrane potential. In rest, m is small and h is large. When the voltage does not reach the threshold, neither m nor the sodium conductance increase. The activation constant corresponding with V_{rest} is small for potassium.
When the current increases, m also increases, while h decays slowly. G_{Na} increases and there are a lot of sodium ions to displace. The consequence of the small time constant for m is that the increase is fast. The time constant for h is much larger and the inactivation of sodium starts much later, but before the peak in the action potential. The time constant for n is large and consequently it takes a relatively long time for potassium ions to flow in opposite direction, out of the cell, causing the end of the action potential. This opposite ion flow causes the repolarization of the cell, even below the resting potential. This is called hyperpolarisation and is caused by the increase of the K conductance and the long inactivation time of K, because of the large time constant n. An action potential can hardly be initialized during the hyperpolarization period. This period is called the *refractory period.*
The *absolute* refractory period is the time interval during which no stimulus is capable of generating another action potential, not even a strong one. The *relative* refractory period is the time interval during which a second action potential can be generated, but it requires a stronger input. The duration of the repolarisation depends on the speed of the Na-channel inactivation and the K-channel activation.

1.7 The Cable Equation

The effect of an injection current is not only measured in the injection point. The potential in all (indirectly) connected positions changes as well.

At the injection position, the potential is maximal. The cable equation describes the attenuation of the voltage when it propagates away from the input site to the dendritic cable. The time development of this voltage is shown in Figure 1.4.

At any point along a cylindrical membrane cable the current is flowing either longitudinally (along the dendrite axis) or back through the membrane.

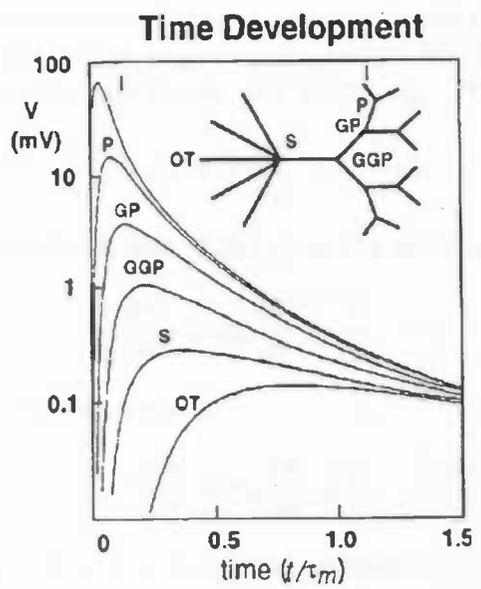

Figure 1.4: The time-course of a voltage change is broadened and the peak is delayed as it propagates away from the input site. A brief transient current is applied to branch I and the resultant voltage at the indicated points is shown on a logaritmic scale.

The current is taken positive when flowing in the direction of increasing values of x. The resistance r_i (in Ω/cm) is the resistance per unit length along the x-axis. With Ohm's law this current is:

$$\frac{1}{r_i}\frac{\partial V}{\partial x} = -I_i, \tag{1.24}$$

where I_i is the longitudinal current.
The current either crosses the membrane via the passive membrane channels which are represented with the resistance r_m (in $\Omega * cm$) per unit length, or (dis)charges the membrane capacitance per unit length c_m (in F/cm). The change per unit length of the longitudinal current, $\partial I_i/\partial x$, must equal the density of the membrane current per unit length, i_m. That is

$$\frac{\partial I_i}{\partial x} = -i_m = -(\frac{V}{r_m} + c_m\frac{\partial V}{\partial t}). \tag{1.25}$$

Combining Equations (1.24) and (1.25) gives the cable equation:

$$\frac{1}{r_i}\frac{\partial^2 V}{\partial x^2} = c_m\frac{\partial V}{\partial t} + \frac{V}{r_m}. \tag{1.26}$$

This equation can be rewritten as

$$\lambda^2\frac{\partial^2 V}{\partial x^2} = \tau_m\frac{\partial V}{\partial t} + V, \tag{1.27}$$

where $\lambda = \sqrt{r_m/r_i}$ and $\tau_m = r_m c_m$ or, normalized,

$$\frac{\partial^2 V}{\partial X^2} = \frac{\partial V}{\partial T} + V, \tag{1.28}$$

where $X = \frac{x}{\lambda}$ and $T = \frac{t}{\tau_m}$.

1.7.1 Steady-State Voltage

When the voltage is constant, $\partial V/\partial t = 0$ and the equation looks like:

$$\frac{dV^2}{dX^2} - V = 0, \tag{1.29}$$

whose general solution can be expressed as

$$V(X) = Ae^X + Be^{-X}, \tag{1.30}$$

where A and B are constants whose values depend on the boundary conditions.

In the case of an infinite cable with $V = 0$ at X $= \infty$ and V $= V_0$ at $X = 0$, then:

$$V(X) = V_0 e^{-X} = V_0 e^{-x/\lambda} \tag{1.31}$$

is a solution of the cable equation.
From the beginning of the cable the voltage attenuates exponentially with distance.
This situation can be approximated in a model of a neuron with an input in the middle of the dendrite, where the influence of the soma and of the end of the dendrite is hardly noticeable.

The dimensionless *electrotonic length* of a finite dendritic cable with length ℓ is defined as $L = \ell/\lambda$.
When no longitudinal current flows at the end of a cylindrical cable with a sealed end at $X = L$, the solution for the cable equation when the voltage is constant and with $V = V_0$ at $X = 0$ is

$$V(X) = \frac{V_0 cosh(L - X)}{cosh(L)}, \text{ for } \frac{\partial V}{\partial X} = 0 \text{ at } X = L. \tag{1.32}$$

The voltage attenuates in this sealed end situation much slower with distance than in an infinite cable.
This situation can be approximated in a model of a neuron with an input at the end of the dendrite, with the consequence that the current only leaks away through the dendritic membrane.
When the point $X = L$ is clamped to a potential, here chosen as 0, the solution is

$$V(X) = \frac{V_0 sinh(L - X)}{sinh(L)}, \text{ for } V = 0 \text{ at } X = L. \tag{1.33}$$

The voltage attenuates much faster with distance than in an infinite cable.
This situation can be approximated in a model of a neuron with a dendrite leading to a large soma. This soma has a large surface where the capacitance is large and much charge is needed to cause a voltage difference. In this situation, the voltage at distance L is (almost) 0.
In Figure 1.5, the voltage attenuation of the different situations is shown.

1.7.2 Electrically small cells

Another special situation of the cable equation considers $\partial V/\partial x = 0$. When V is a constant and there are no voltage differences the cable is an isopo-

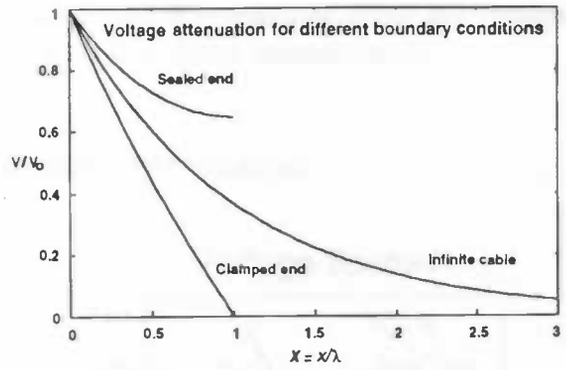

Figure 1.5: Attenuation of membrane potential with distance in a cylindrical cable with different boundary conditions. The middle curve shows the attenuation in an infinite cable. The other two plots are a finite length cable of electrotonic length L=1. The upper one is for a sealed end cable and the lower is for a cable with the end clamped to the resting potential of 0.

tential element. The cable equation is reduced to an ordinary differential equation,

$$\frac{dV}{dT} + V = 0, \tag{1.34}$$

with solution

$$V(T) = Ae^{-T}. \tag{1.35}$$

A depends on the initial condition.
This situation is a good approximation of small neurons without large dendrite trees, for example only a single soma. These cells are called *electrically small cells*. A current step pulse injection I_j to the element causes the voltage to behave like mentioned in Equation (1.7):

$$V(T) = I_{inj}R_m(1 - e^{-T}) = I_{inj}R_m(1 - e^{-t/\tau_m}), \tag{1.36}$$

R_m being the membrane resistance of this element.
When the injection stops at $t = t_0$, the voltage decays exponentially from its maximal value $V_0 = V(t_0)$ according

$$V(T) = V_0e^{-T} = V_0e^{-t/\tau_m}, \text{ for } t \geq t_0. \tag{1.37}$$

V_m decays exponentially at the end of the injection current. Each τ msec the voltage decays with a factor $e^{-1} = 0.37$. With a dendrite added to the

soma, the synaptic potential propagates along the membrane and arrives broader and with a delay in other compartments.

1.7.3 Asymmetric Attenuation

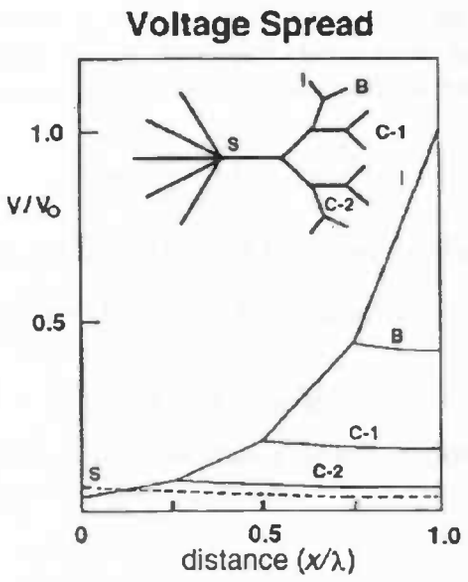

Figure 1.6: The voltage spread in passive dendritic trees is assymetrical. The solid curve shows the steady-state voltage computed for current input to terminal branch I. Large attenuation is expected in the input branch whereas much smaller attenuation exists in its identical sibling branch B. The dashed line corresponds to the same current when applied to the soma.

The voltage attenuation in the central direction and the periferal direction is asymmetric. A current flow in the central direction (from dendrites to soma) is the "voltage clamp" situation and a flow in the peripheral direction (away from soma) is a "sealed end" situation.
There is more current leakage in the direction dendrites to soma and, consequently, the voltage decays much faster in that direction. Another way to say this is that the dendrites are rather small from the soma point of view, while the soma is large for the dendrites. Figure 1.6 illustrates this point.
The consequence of this fast attenuation in the direction dendrite to soma

is that more excitatory inputs at distal positions than at the soma lead to an action potential.

1.7.4 Input Resistance

An important parameter for a measure of the responsiveness of a specific region to its inputs is R_{in}, the axial *input resistance*. R_{in} is equal to the value of the ratio V_0/I_0, where I_0 is the steady state value on a certain location and V_0 the steady state voltage developed there. In an infinite cylinder the steady current input must split into two directionss from injection point *x=0*. Thus

$$I_i = -\frac{1}{r_i}\frac{\partial V}{\partial x}\bigg|_{X=0} = \frac{I_0}{2}. \tag{1.38}$$

From Equation (1.31) we know that $(\partial V/\partial x)|_{X=0} = -V_0\lambda$. This leads to

$$R_{in} = V_0/I_0 = r_i\lambda/2 = \sqrt{r_m r_i}/2, \tag{1.39}$$

or

$$R_{in} = (1/\pi)d^{-3/2}\sqrt{R_M R_A}. \tag{1.40}$$

This means that thinner cylinders have a larger R_{in} compared to thicker cylinders with the same R_A and R_M values.

1.8 τ and λ

A measure of the duration in which an input is noticeable is τ_m. The value of τ_m depends on the electrical properties R_M and C_M of the membrane, $\tau_m = R_M C_M$. A cell with a few open membrane channels (large R_M, large τ_m) remembers the effect of the input longer. Neurons with a small τ_m respond quickly to the input, but forget it rapidly as well.
Another name for τ is time or integration constant, referring to the period the neuron can summate inputs. A neuron with a large τ summates inputs over a long period of time.
The effect of the previous inputs is still noticeable and the effects of the inputs integrate.

A measure for the attenuation as function of distance is the space constant λ, which not only depends on the membrane properties but also on the axial resistance and the diameter of the dendrite,

$$\lambda = \sqrt{(d/4)R_M/R_A}. \tag{1.41}$$

With larger λ, the voltage attenuation reduces less with distance. As a consequence, the input is noticed more distantly. Inputs distant from each other will summate better (spatially) when λ is large.
The value of λ depends on the ratio of membrane resistance and axial resistance. When R_M is large and R_A small, little charge leaks away through the membrane and the effect is noticed more distantly.

Chapter 2

The Model

2.1 The Choice of Genesis

Two alternatives to simulate a neuron are to do the calculations analytical, by hand, or numerical, by a program.
Calculating by hand is hard, because there are a lot of formulae to integrate. And apart from that, performing the non linear summations by hand is difficult.
The other possibility is to let a program perform the calculations. A potential problem when a program performs the calculations is that it is not completely sure that it calculates the right things. This is also true for calculations by hand. But the numerical model can be gauged with calculations by paper and pencil and by the literature value ranges. Consequently, the gauging is important.
I already knew how to work with Genesis, the GEneral NEural SImulation System (Bower, Beeman, 1997). Genesis is a graphically oriented general purpose numerical simulator to facilitate modelling of neural networks. The simulator is designed to support simulations at many levels of detail. Specifically, it is intended for use in both applied network modelling and in the simulation of detailed, realistic, biologically-based models.

The neuron to be modelled has one soma and two dendrites. This neuron combines information from the right and the left ear. It must be possible to provide multiple inputs to one dendrite to simulate spatio temporal effects. It also must be possible to generate input patterns and provide them to the neuron. And last, but not least, the model must be realistic with realistic parameter values and behaviour.

"The best place to start a new modeling project is with an existing

model, but which model and at which level[1]?
I decided to use the existing cable model which models a dendritic cable with a specified number of compartments and calculates the membrane potential for each compartment. It is possible to provide an injection current or synaptic input to any particular compartment. I adapted this model until it had all the required properties.

2.2 Working with Genesis

2.2.1 Initialisation

The simulator consists of graphical interface tools and a scripted language. In the framework of the Script Language Interpreter (SLI) the programmer defines and manipulates GENESIS elements, arithmetic operations and conditional statements. A single simulation consist of a number of scripts. The SLI uses these scripts to combine pieces of a simulation. The commands are executed by the SLI either from the commandline or from script files (sequences of script commands are stored in files). The system's available resources consist of module libraries and the simulator base code. The user extends the library files, with a large base of routines and modules to design a simulation, in the base code, which performs all the setup and control instructions specified by the interface. The base code also handles all the timing and update sequencing for the modules, and provides access to the variables in the simulation.
Working with modules is flexible and easy, because an alteration or addition to an existing program requires only changes to or addition of a few discrete modules.

To design a particular simulation, the user must first select the required modules from the libraries and link them to the base code.
There are three sorts of modules used in Genesis: computational, communications and graphical.

- Computational modules
  Computational modules are called elements. Elements are the building blocks of simulations. All the calculations are performed by elements, created from templates called "objects". An object contains specific information needed to construct a particular type of element. An object can perform several operations on its data and is attached to compiled functions in which the operations are defined. A module

[1]Book of GENESIS, Bower and Beeman, page 199

is written with options to update functions and to assign values to parameters, which is performed by the script language.

- Communication modules
  These modules are called messages and connections and are used to link the elements. The design of a simulation requires the creation of many elements. Initially the elements are not linked. Messages link the elements so that the correct computational units are formed. The linking is performed before the execution, so messages do not have a time delay. After the initialisation, the element field values that are linked by messages are updated every, with user installable, time step. Connections interconnect the computational units. They, for example, perform the function of axons providing time delays and synaptic strengths.

- Graphical modules
  Graphical modules are called widgets and are needed to construct the interface, provided by Xodus, the X-windows Output and Display Utility for Simulations. Xodus is independent of Genesis, but the created non-computational object oriented elements are treated like Genesis elements. Like the other modules, they depend on the script interface for most of their interaction with the rest of the simulator. Widgets can either issue script commands or respond to them.

When the modules are linked to the base code, the simulation is set up by writing a sequence of commands to establish the neural network itself and the graphical interface. Simulations are constructed of many different types of elements with several relationships, mediated by messages and connections.

The structure of the element creation is hierarchical (just as directories in UNIX). Messages and connections are linked to elements as files are to directories. When an element is created, it must be linked to an already existing element, lower in the tree.

Each element has an associated interval timer or clock, used to determine how frequently the element will be executed during the simulation. Usually the clock is the global simulation clock, clock number 0, with the basic simulation time step. Other clocks are useful when an element runs at a significantly different time scale. The user sets the value of the simulation time step of the clock, and the simulator updates the values of the specified elements each time a step is performed.

2.2.2 Execution

Before running a simulation, the elements must be initialised with the `reset` command. Then messages are sent and values are assigned to the parameters.

Together with the simulation code, the program creates a list with the operations, performed by the simulator in the specified order. With each time step, the program works through the list. The list is called a *schedule* and the operations are called *tasks*. A task is a compiled function with optional arguments. An example of a task is to check for any inconsistencies in the model. This task should be performed after each change of the schedule. Another task is the simulation of the model elements.

So the only thing the program does during the simulation is calculating the values of the message fields of the initialised messages. These calculations are performed every simulation step.

2.3 Designing the Model

2.3.1 Soma and Dendrites

An object of the type `neutral` is an empty element that performs no actions and is used mainly as a parent element for a hierarchy of child elements. The model generates the neuron compartments as children of this neutral object. Genesis implements a dendritic cable model as an equivalent cylinder. It creates a one-dimensional compartmentalized cable as shown in Figure 2.1. The cable consists of a soma compartment and N cable-compartments. N is variable and can be set to any desired number. The parameter values may also be different per compartment. The soma is a cylindrical compartment, just as the dendrite, but the soma is typically bigger than a dendrite compartment.

A cylindrical compartment is similar to the one shown in Figure 1.2). The object `compartment` simulates such a section of a passive membrane or cable. The potential across the membrane is given by V_m. There is a leakage current through the resistance R_m. This resistance is put in series with a leakage potential E_m. This compartment can be linked to other compartments with an axial resistance R_a. The compartment is not symmetrical, with R_a lumped to one side of the compartment. Any number of ionic channels can be introduced into the membrane (G_k, E_k in the circuit diagram). The membrane also allow current injections. For the calculation of V_m for

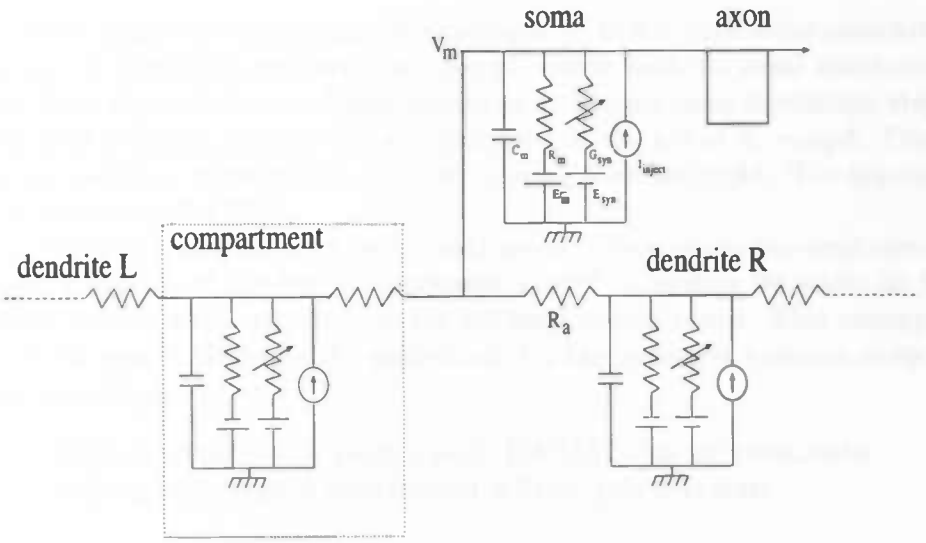

Figure 2.1: The electric circuit of the simulated neuron

one compartment, the next formula is used:

$$C_m \frac{dV_m}{dt} = \frac{(E_m - V_m)}{R_m} + \sum_k [(E_k - V_m)G_k] + \frac{(V_m' - V_m)}{R_a'} + \frac{(V_m'' - V_m)}{R_a} + I_{inject}.$$

V_m' and V_m'' represent the membrane potentials in the adjacent compartments. Here, the sum over k represents a sum over the different types of ion channels that are present in the compartment. The variable conductance of each channel type G_k gives the net effect of many individual channels that open and close in a binary manner.

When a compartment is created, the fields E_{rest}, *length*, *diameter*, R_m, R_a and C_m are set to the installed values. Initially the menus forced the user to fill in values for R_M, R_A, C_M, *length* and *diameter*. But it is easier and more insightful to do this in terms of the characteristic length and time constants of the dendrites and inputs, because that is what this model is about. So the menus were changed and the user can now choose values for τ, λ and L, which were explained in Section 1.8.
These characteristic properties are used different for the soma. Although the soma is a sort of compartment in this model (with larger values for length and diameter), a definition in terms of L is not useful, because there is only a single soma, independent of the dendrites. For the same reason, speaking of λ is not useful either.

Each compartment requires two messages to link it with other compartments. A dendritic compartment, *comp1*, sends both its axial resistance (the field R_a) and its membrane potential at the previous simulation step (the field *previous_state*) to the compartment at the left of it, *comp2*. That allows *comp2* to calculate the current entering from the right. The type of this message is RAXIAL.
The righthand compartment *comp1* only needs to receive the previous membrane potential of the left compartment *comp2* to update its state, as it knows its own axial resistance to the lefthand compartment. This message is of the type AXIAL. So the statements for the messages between *comp1* and *comp2* are

```
addmsg path/comp1 path/comp2 RAXIAL Ra previous_state
addmsg path/comp2 path/comp1 AXIAL previous_state
```

2.3.2 Synapses

A synapse is representated as a time dependent synaptically activated channel. This **synchan** object must be connected to the dendritic compartment. Because the model requires a variable input, where the parameters can differ per compartment, the number of synaptic channels must equal the number of compartments (plus one for the soma, so an input can also be provided to the soma).

The **synchan** object may receive delta-function "spike-events", each lasting for a single integration time step, from a SPIKE message. It then calculates a net channel conductance G_k that sums the effects of each spike. To calculate the temporal behaviour, this objects uses the alpha-function, see Section 1.3.2, with the parameters g_{max}, τ_1 and τ_2. Usually τ_1 is equal to τ_2, which causes G_k to reach a maximum value of g_{max} after a time τ_1, because

$$G_k = g_{max} \frac{t}{\tau_1} exp(1 - \frac{t}{\tau_1}).$$

The initial rise in conductance is approximately linear and the decay is approximately exponential, with time constant τ_1.

Each SPIKE message to a **synchan** establishes a synaptic connection and increments **nsynapses**. The synapses are numbered, starting with 0, and each synapse contains a field for the synaptic weight and a propagation delay. For example, the weight of the first synaptic connection is held in the field *"synapse[0].weight"* while the delay is *"synapse[0].delay"*. G_k reaches a value $g_{max} * weight$ for a single event delivered with a SPIKE message. Any

number of spike events can be pending per synapse. The object **synchan** stores spike events in a buffer until they are scheduled to occur.

Apart from the need for a function that sets the fields to the desired values, there has to be a function that links the synaptic channel to the associated compartment. This is, again, implemented by passing messages. The compartment requires the conductance and the equilibrium potential of the channel. This is used by the compartment for the calculation of the net current flow into the compartment. Altough the channel conductance is independent of the membrane potential, the **synchan** object also calculates the channel current, consequently it requires a message from the compartment providing the membrane potential. The statements look like:

```
addmsg compartment/channel compartment CHANNEL Gk Ek
addmsg compartment compartment/channel VOLTAGE Vm
```

2.3.3 Widgets

To inspect the behavior of the simulation and to modify its parameters, an interface is designed. The control panel has buttons that pop up menus in which parameters can be modified. the interface is shown in Figure 2.2.
In the first place the buttons to reset, stop, quit and run the simulation were made. Within the button for running there is a possibility to set the simulation time. The integration time step determines how frequently the calculations will be updated during the simulation.

Other buttons specify the desired number of compartments and the compartments to be plotted. Furthermore, there are buttons to modify the cell parameters, the synaptic input and the current injection and one button to read input from a file. When a button is pressed, the function belonging to it is called, for example to set a certain value or to add a compartment plot to the graph.

The user has to make sure that, in the compartment in which the input has to be provided, the toggle is set to ON. If not, the messages are not executed and no input is given.

Two graphs show the membrane potential and the log membrane potential of the different compartments and the soma. Another graph shows the conductance in the soma. Because all the compartments of the neuron are connected to each other, the graph requires only a single message from the soma to calculate the potential in all the compartments.

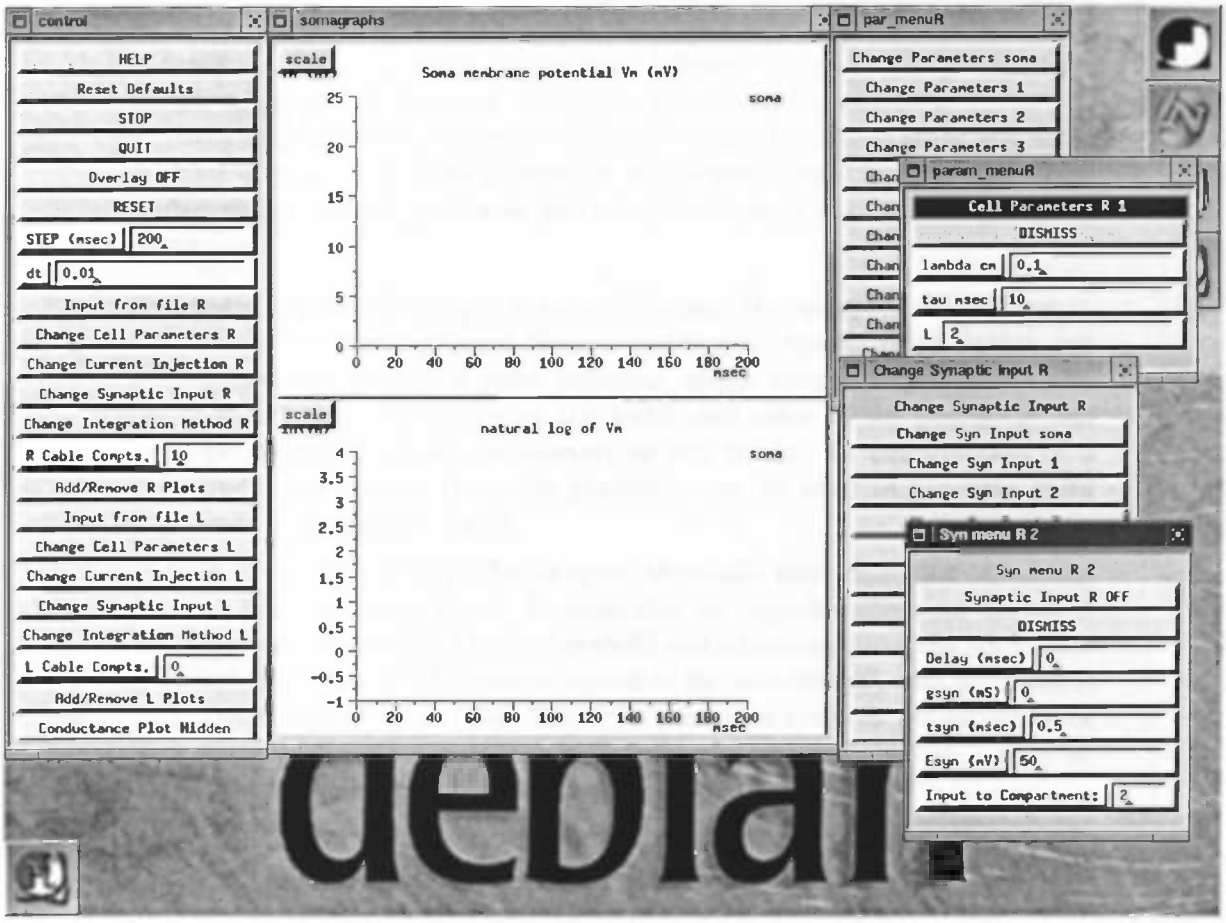

Figure 2.2: The interface.

2.3.4 Input

Providing an input to the model requires objects that generate input pulses or spikes.

Initially there was just a single pulse generator and a single spike generator. When all compartments were linked, they all got the input from a single generator. In that way it is possible to provide multiple inputs to different compartments, but because the inputs all stem from one generator it is not possible to provide inputs with different values for the parameters. These values are all the same. Only the gain could be modified.

Providing different inputs to different compartments requires the same number of pulse and spike generators as compartments, so that each compartment has its own generator and synaptic channel to provide any desired input.

Synaptic input. To supply a synaptic input, the synaptic channels must receive activating input from a `pulsegen` object. This device can generate a variety of pulse patterns: single pulses, double pulses and pulse trains, depending on the width and delay of the pulses. It can be triggered, gated, or allowed to run freely. In combination of a `spikegen` object, the pulse generator can be used to generate spike bursts as synaptic input.

A spike generator performs spike threshold discrimination according to the Genesis manual. It generates an impulse whenever an input exceeds the specified spike threshold and whenever there has not been a spike for at least the interval specified by *'abs_refract'*, that determines the maximal spike rate. The spikegenerator receives potentials from a compartment via the INPUT message and sends spike events to synaptic channel elements with a SPIKE message.

The pulse generator is set in such a way that it produces only a single long pulse. This means that the width of the pulse and the delay of the second pulse are set to a value higher than the simulation time. This pulse is the input of the spikegenerator. The input is always either 0 or 1, so the threshold is set to 0.5. If *abs_refract* is set at a large value so that precisely one spike is produced and not a whole spiketrain.

When the input is 1 and a spike is generated, the SPIKE message is sent to the synaptic channel, which is set to a desired reversal potential, maximal conductance and pulse width. This channel is connected to the neuron as was explained earlier. The neuron integrates the

synaptic input with the other messages to compute the membrane potential and the channel conductance.

Injection current. An injection current is implemented similarly. Again, first a `pulsegen` object is combined with a difference amplifier to generate an injection current as input.

According to the genesis manual the object `diffamp` is a difference amplifier which takes two inputs and produces an output proportional to their difference. It adds inputs from PLUS messages and subtracts those received with MINUS messages. The output is this total, multiplied by specified gain, but limited to the specified range -saturation to +saturatation. If there are no MINUS messages, the minus input is taken as zero.

The pulse generator is, again, set to produce a single long pulse, which is send with a PLUS message to the difference amplifier, where the parameters *saturation* and *gain* are set to desired values. The `diffamp` object calculates the output and sends this, by using an INJECT message, to the soma, which integrates this message with the other messages and performs the calculations.

Input from a file. A more elaborate synaptic input is supplied from an input file. The compiler reads at each clock tick when and where an synaptic input must be supplied. The input file exists of a sequence of 0's and 1's, (one for each segment) separated by spaces.

The `disk_in` object reads in from an input ascii file which has a two dimension array for values, to fill the *val[x][y]* array. The two dimensions are x and y, x the number of rules to read in one clock tick and y the number of data (here compartment) elements.

The array with different clocks is used to let the `disk_in` object run at a time scale with a greater interval then the basic simulation time step.

For each compartment a `spikegen` object is implemented after creation and parameter setting of the *disk_in* object. After that, an INPUT message is sent from the `disk\_in` to the `spikegen` object. When the input is 1, a SPIKE message is sent from the `spikegen` to the associated `synchan`, which is linked to the associated `compartment`. All messages for a synaptic input are sent and the neuron integrates these with the other messages and performs the calculations.

2.3.5 The Program Structure

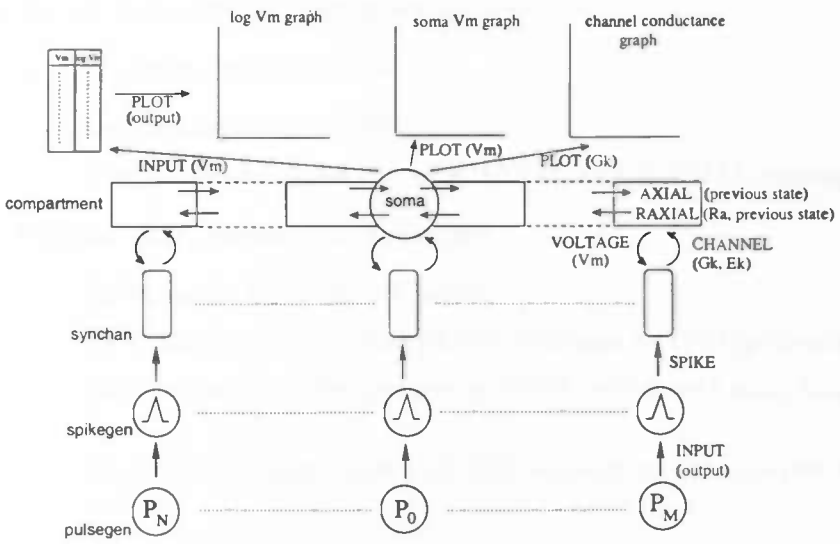

Figure 2.3: The program structure.

The program structure of the simulated neuron is shown in Figure 2.3.

2.4 Overview of Initialisation

The next list shows all the steps the model takes to initialize a synaptic input.

- the soma is made
- and the pulsgenerator and the spikegenerator are added
- *delay*, *width* and *interval* of the input are set
- the pulsgenerator sends an INPUT message to the spikegenerator
- all graphs and menus are loaded
- the soma sends a PLOT message to the somagraph (initialisation of the graph)

When the function `add_cable` is called from the widget menu the following actions are performed:

- the old cable (if exists) is deleted
- the old messages and menus are deleted
- function `make_cable` is called
  - compartments are added
  - compartments are linked with AXIAL and RAXIAL messages
- function `make_param_cable` is called
  - the synaptic channels are added
  - the synaptic chanels send PLOT messages to the G_k-channel
  - the function set_chan_params is called, which sets g_{syn}, t_{syn} and V_{syn}
  - the spikegenerator sends a SPIKE message to the synaptic channel
  - *delay* and *weight* of the synaptic channel are set
  - a synapse menu is made for each compartment with the possibility to
    - switch the toggle between ON and OFF
    - modify the values for *delay*, g_{syn}, t_{syn}, V_{syn}
  - a parameter menu is made for each compartment with the possibility to modify
    - R_A, R_M, C_M
    - soma and compartment *length* and *diameter*
- the soma is linked to compartment 1, and with this to the rest of the cable
- all old cable plots are removed
- reset, which means that all elements are placed in the known initial state
- echo: N passive cable compartments added R/L of soma

When the function `add_syn_input` is called from the widget menu the following actions are performed:

- if toggle is ON the function `link_channel` is called

  - the synaptic channel sends a CHANNEL message to the associated compartment
  - the compartment sends a VOLTAGE message to the associated synchan

- else, toggle is OFF, the function `unlink_channel` is called
  - there is a check if connections by messages exists between the synaptic channel and the compartment that has been called
  - if that is so, these messages are deleted

Chapter 3

Simulations

3.1 Validating the Model

3.1.1 Robustness

The simulated neuron varies with different values of τ and λ. The anatomic values for the parameters fall within particular ranges. The values for λ and τ were varied within these default ranges. If a synaptic input is provided to the neuron, the resulting conductance is in the form of an alpha-function, with a peak value (g_{max}) and a time-to-peak (t_p). The variations in the peak potential V_{peak} and the width of the potential at $0.8 * V_{peak}$ were measured. The width was measured in *msec*, by measuring the peak potential V_{peak}, calculating $0.8 * V_{peak}$ and measuring the times of the rise and decay of the potential at that potential.
Pulses, all of $1e^{-}5$ mS were provided at compartment 10 (at the end of the dendrite) and to the soma. All values were measured in the soma.

The Figures 3.1 to 3.4 show the variations in the potential and width. One property is varied, the others remain the same.
Doubling these values does not lead to large changes in potential or width. There only is a significant difference in the width of the potentials when the input were provided in compartment 10 and τ was varied.
From this study, we can conclude that the model is quite robust for variations in τ and λ.

3.1.2 Realistic values

In this robustness study, the physiological parameter values of R_m, R_a, C_m, etc. change due to the variation of τ and λ. These parameter values fall

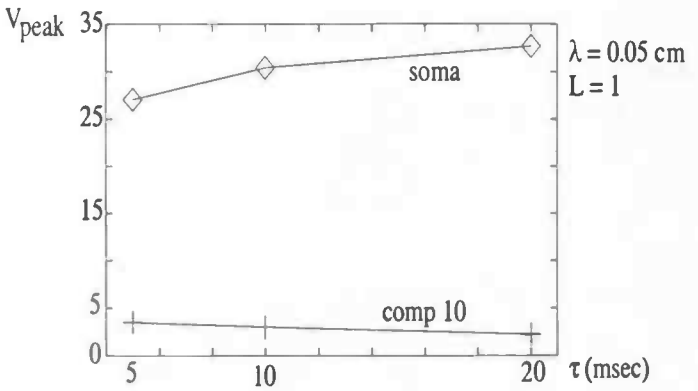

Figure 3.1: The variation in the peak potential when varying τ.

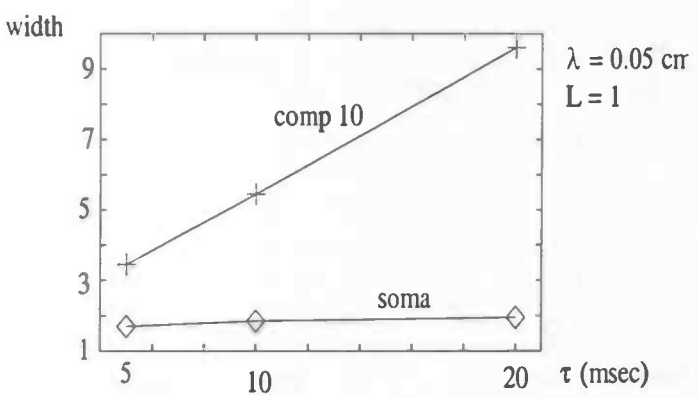

Figure 3.2: The variation in the width of the potential when varying τ.

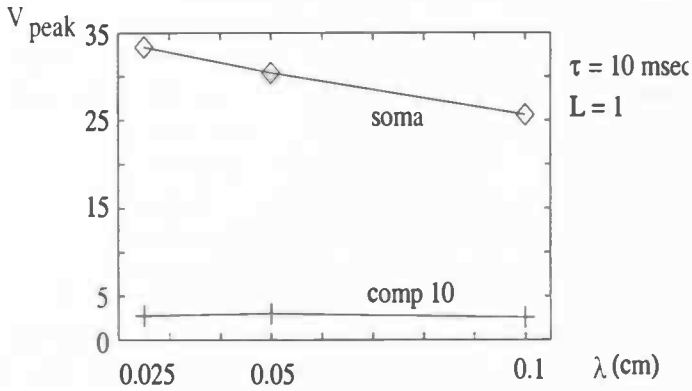

Figure 3.3: The variation in the peak potential when varying λ.

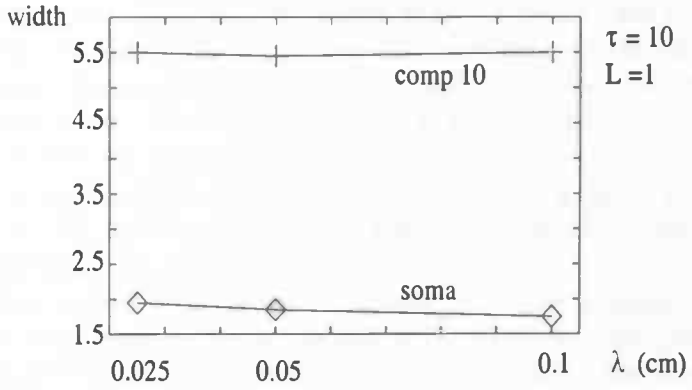

Figure 3.4: The variation in the width of the potential when varying λ.

parameter	literature value	used value	formula
R_A	70-250 Ωcm	25-800 Ωcm	see λ
R_M	5-50,000 Ωcm^2	3,333-13,333 Ωcm^2	see τ
C_M	1-2 $\mu F/cm^2$	1.5 $\mu F/cm^2$	see τ
dendrite length	1-10 mm	0.13-3 mm	
dendrite diameter	1-6 μm	1.5 μm	see λ
λ	0.2-1 mm	0.25-1 mm	$\lambda = \sqrt{(d/4)R_M/R_A}$
τ	7-50 ms	5-20 ms	$\tau = R_M C_M$
L	0.2-2	0.5-2	$L = x/\lambda$
t_{peak}	0.3-1 ms	0.5 ms	
g_{peak}	0.1-1 nS	0.31-10 nS	

Table 3.1: Parameter values known from the literature of anatomic studies compared to the used values in the simulated model.

within ranges, known from the literature of anatomic studies. Within these ranges, one can choose the values. The neuron was designed so that the effects of the stimulations were optimal, while most of the parameter values fell within the default ranges. The list is shown in table 3.1.

Some parameter values fall outside the ranges. This is possible, if a justification can be provided.

The length of the used dendrite is small. The length is defined as $\ell = L * \lambda$. The maximal value for L is *2* and the maximal value for λ is *1 mm*. The consequence is that the maximal length is *2*1=2 mm*. This looks like an inconsequence in the literature values. If the length of the dendrite was chosen longer, the potential in the soma should be very broadened, not useful for this study. So the used dendrite is not very long, but still longer than the one used by Agmon-Snir et.al.

The value of R_a could become quite large, because a small value can be chosen for λ. R_a could become quite small, because the diameter of the dendrite can be chosen small.

A small value is used for the time constant τ. When a synaptic input is provided to compartment 10 (at the end of the dendrite), the peak is small and broadened when arriving at the soma. If τ is chosen a few *msec* smaller, the peak is slightly broader.

The value for g_{peak} is large, compared to the literature value. The reason for this is the same as for using a small τ. The peak of an input provided to compartment 10 is small and broadened when arriving at the soma. If a small input is provided, almost nothing is left in the soma. In reality, there are more synapses at the end of the dendrite than at the beginning. Using

this fact, a larger input was provided to compartment 10.

From this study, we can conclude that the freedom to choose parameter values is quite large for most of the parameters, while for all that there is only little freedom to keep all the parameter values at the same time within the ranges of the literature of anatomic studies.

3.1.3 Validating on τ

In the situation where $\partial V/\partial x = 0$ for each X, V is constant. This means that there are no voltage differences: the cable is an isopotential element. This situation is a good approximation of a single soma and is used to gauge the model on τ. The cable equation, Equation 1.28 reduces to:

$$V(T) = -\frac{dV}{dt}. \tag{3.1}$$

Given a constant injection input I_{inj}, starting at $t = 0$, the solution is:

$$V(T) = V_0(1 - e^{-T}) = I_0 R_m (1 - e^{-\frac{t}{\tau_m}}), \quad \text{with } 0 < t < t_0. \tag{3.2}$$

During the application of a positive current pulse to the interior of the cell,

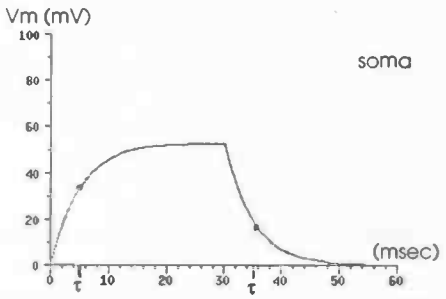

Figure 3.5: The validation of τ. The time $t = \tau$ is equal to the time it takes the voltage to rise to $1 - e^{-1}$ of its maximal value and the time it takes the voltage to decay to e^{-1} of its maximal value.

the membrane potential depolarizes exponentially from the resting potential (0) to the maximal value $I_{pulse}R_m$, according to Equations 1.7 and 1.36. The rise is governed by the single time constant τ_m, which is equal to the time it takes the voltage to rise to 63% $(1 - e^{-1})$ of its maximal value.
This holds in the simulated model, with the time constant τ_m taken as 5

msec. The maximal value is *50.7* mV and the voltage at $t = \tau_m = 5$ *msec* = *32.2* mV. Figure 3.5 shows that at time $t = \tau_m$, the voltage is increased with $\frac{32.2}{50.7} * 100\% = 63\%$ of the maximal value.

The exponential decay is also governed by τ_m. At $t = t_0$, when the current approaches V_0, the pulse ends and the voltage attenuates exponentially to 0 with, see Equation 1.37:

$$V(T) = V_0 e^{-T} = V_0 e^{-(t-t_0)/\tau_m}, \quad \text{with } t_0 < t < \infty. \tag{3.3}$$

This also holds in the simulated model. At $t = 35$ *msec*, where $t - t_0 = 5msec$, the voltage is 18.7 mV. Figure 3.5 shows also that at time $t - t_0 = 5$ *msec*, the voltage is attenuated with $\frac{18.7}{50.7} * 100\% = 37\%$.

3.1.4 Validating on λ

The validation of λ is performed by an approximation of an infinite cable, to investigate the transport through a relatively long dendrite (in terms of λ).
A constant input is provided to one side of the cable. In the steady-state, V(t) is constant and the cable equation, Equation 1.28 reduces to:

$$\frac{d^2}{dX^2} V(X) = V(X). \tag{3.4}$$

The boundary conditions of an infinite cable are: $\lim_{x\to\infty} V = 0$ and $V = V_0$ at $X = 0$. The solution is:

$$V(X) = V_0 e^{-X} = V_0 e^{-\frac{x}{\lambda}} \tag{3.5}$$

The voltage decays exponentially with distance and λ is a measure for the speed of decay.

This holds in the simulated model. In this experiment, the cable has 30 compartments, each with length *0.3* *cm*, so that *L=3*. The soma gets a diameter of $25e^{-4}$ *cm* and a length of $50e^{-4}$ *cm*. An injection current of *0.0002* μA is provided for 100 *msec* to compartment number 20, at $L = 2$. The membrane potential is measured at *100* *msec* in compartment 20 (at $L = 2$) and compartment 10 (at $L = 1$). In compartment 20, the potential is *31.6* mV and in compartment 10 *11.3* mV, see Figure 3.6.
The charge is transported over $L=l/\lambda=1$. *V(X)* must be attenuated to *37%* in compartment 10. With the measured data this is true: $\frac{11.3}{31.6} * 100\% = 37\%$.

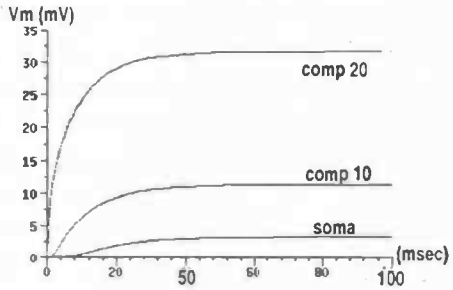

Figure 3.6: The validation of λ. In a distance $L=1$, the voltage is decreased to a factor e^{-1}.

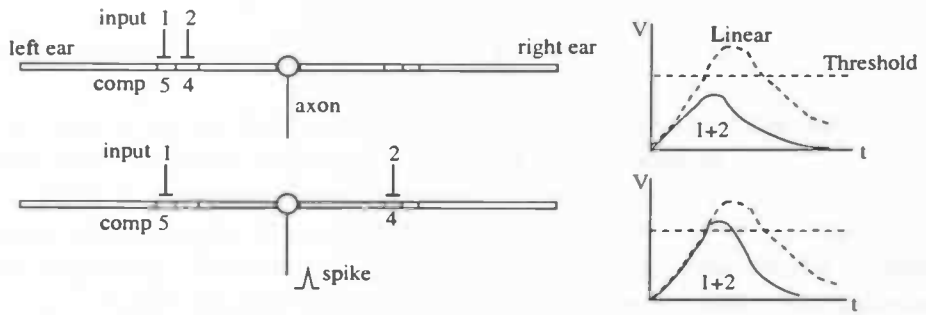

Figure 3.7: The effect of non-linear summation. Two inputs provided to one dendrite does not lead to the generation of an action potential, while providing one input to both dendrites does.

3.1.5 Bipolar Dendrite

Synaptic inputs can be activated either close together, at the same dendrite, or at different parts of the dendritic tree, at two different dendrites. In the former case, the local voltage change must be larger than in the latter and the reduction in the driving force for the synaptic current is more pronounced. This is the consequence of nonlinear summation. In the latter case, more depolarizing current is generated by the synapses and a spike is more likely to be generated in the axon, because nonlinearity is less significant.
Figure 3.7 illustrates these situations.

To prove this non-linear summation in the simulated model, ten compartments are added to the left and to the right of the soma. Synaptic inputs of $1e^{-4}$ mS, 0.5 $msec$ and reversal potential of 50 mV are used. Such an input provided to compartment 4 results in a soma membrane potential of 9.21 mV. Provided to compartment 5, it results in 6.68 mV. The sum of these two inputs is 15.89 mV, which is above the threshold for firing, 15 mV.
One input provided to compartment 4 and one to compartment 5 of the same dendrite results in a soma membrane potential of 11.35 mV, which is below the threshold.
One input provided to compartment 4 and one to compartment 5 of the other dendrite results in a soma membrane potential of 15.79 mV. The threshold is reached now, so a spike is generated.

3.2 Two Inputs, One Peak

The idea is to use the traveling time of the dendrite to investigate if the frequency sensitivity improves when using a long dendrite.
An input is provided to compartment 10 (at the end of the dendrite). It takes some time for the response to travel from the end of the dendrite to the beginning. This time is defined as the traveling time of the dendrite (ΔT_d). Figure 3.8 gives an illustration. The pulse spreads when traveling through the dendrite, what is called *dispersion*.
Another input is provided to compartment 1 (next to the soma).
The interstimulus time ΔT_s is the time between the providing of the pulses.

If ΔT_s corresponds to ΔT_d, the response is optimal, because the response peaks coincide. Both inputs sum (nonlinearly) and enter the soma at the same time. An action potential might be generated, which could not be generated when the times did not correspond exactly.

An input provided close to the soma results in a large potential. The

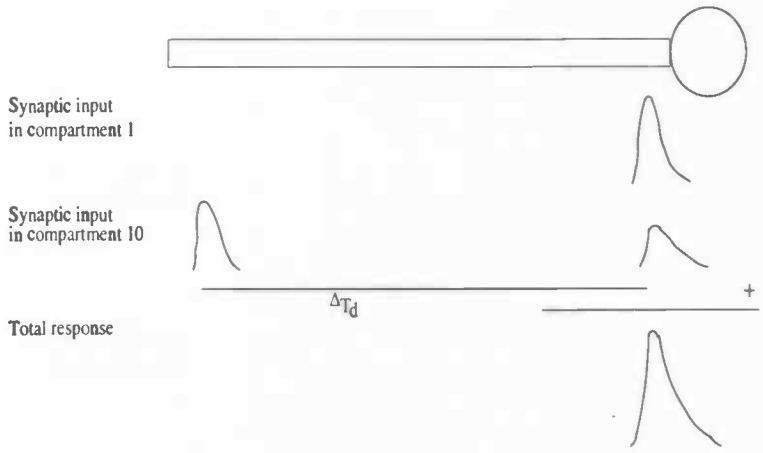

Figure 3.8: An illustration of the traveling time of the neuron and of the optimal response when this traveling time corresponds to the interstimulus time.

traveling way is short and the potential is not much attenuated. An input provided to the end of the dendrite has a smaller influence. The traveling way is long and the potential is much attenuated when arriving at the soma. The output is designed in such a way that a larger stimulus is provided to compartment 10, resulting a maximal potential of about *10 mV* in the soma. The stimulus provided to compartment 1 is smaller, resultinging a maximal potential of about *5 mV*. Together, the responses could generate an action potential (the threshold is *15 mV*).

3.3 A Pulse Train

This response of two inputs is extended to a pulse train. Synaptic inputs are provided to compartment 10 and to compartment 1 at the same time. ΔT_s is the time between the pulses. Figure 3.9 illustrates this. The responses do not arrive at the same time at the soma, because the traveling time of the response of compartment 10 is longer.
If the interstimulus time ΔT_s is changed, so that it corresponds to the traveling time of the dendrite (ΔT_d), the response is each peak maximal.

Again the response is designed in such a way that the maximal response of the input provided to compartment 10 is about *10 mV* and that of compartment 1 about *5 mV*.

In Genesis, it is possible to show the responses of the different compart-

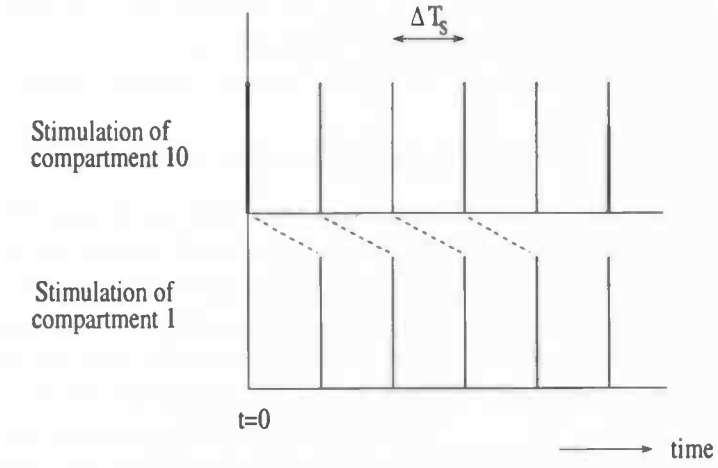

Figure 3.9: The generation of a pulse train by providing inputs to compartment 1 and 10 at the same time, with an interstimulus time between the stimulations.

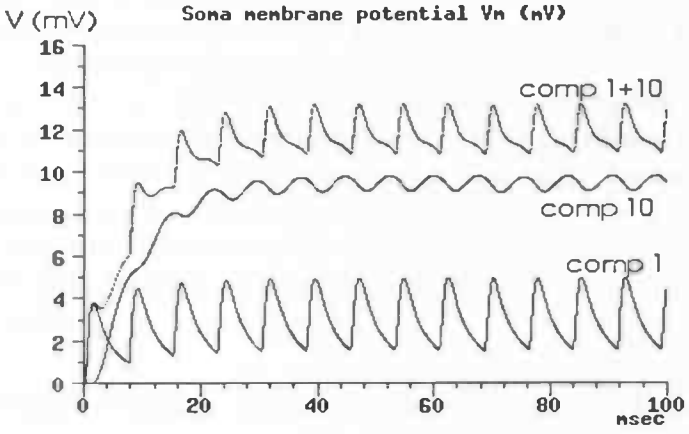

Figure 3.10: The situation when the responses of the peaks do not coincide. Comp 10 is the response of the input provided to compartment 10. Comp 1 is the response of the input provided to compartment 1 and comp1+10 is the combined response.

ments seperately. When using this possibility, one can see if the responses of the inputs in compartmetn 10 and compartment 1 corresponds. If they do not correspond, two peaks arises where there should arise one when the response peaks coincide. Figure 3.10 shows these not corresponding peaks.

3.3.1 The Problem caused by Dispersion

Figure 3.10 also shows that the response of the input provided to compartment 10 is not sharp. The peaks can hardly be called peaks.
The reason is that the dispersion of the input in compartment 10 becomes large, causing the responses do overlay and integrate. And the responses come from the end of the dendrite, resulting in even more dispersion. The pulses cannot be seperated any more, and all effects of an eventual coincidence of the response peaks are lost.

Providing an additional input in the middle of the dendrite to simulate a harmonic, is not an option any more. If this input is provided, the influence of the dispersion should become even larger and the peaks should overlay even more.

I tried to minimize the overlay and integration effects by reducing the time constant τ. This results in a faster rising and falling of the potential. Unfortunately, this reduction leads to a different traveling time. If the interstimulus time is also changed, so that it corresponds to the new tuning frequency, the problem returns; the responses overlay again. Consequently, changing τ cannot solve the problem.
Another problem are the available frequencies in the model. Because the literature gives the maximal value for the length of the neuron as $\ell = L * \lambda = 2 * 1 = 2\ mm$, there is also a maximal traveling time. Because the length cannot increase much, neither can the traveling time. This is a problem because for speech realistic traveling times for the harmonics are in the range of 2 to at least 13 *msec*. The maximum value in the simulated model is about 8 *msec*.

3.4 Pulse Trains with Other Stimulus Frequencies

Because the designed neuron is sensitive to a particular frequency (corresponding to the traveling time of te dendrite), it must react less and generate less spikes, when another stimulus frequency is provided.

3.4.1 Lower Frequency

With a lower frequency than the preference frequency, the interstimulus time is larger. the expectation is that the response is not optimal, because the response peaks do not coincide.
The activation is smaller and the number of spikes is less, because the excitation is smaller as a consequence of a lower stimulus frequency. The number of generated action potentials is less, only because the frequency is lower, and not because the response peaks did not match.
Figure 3.11 shows the response of using a lower frequency than the preferred

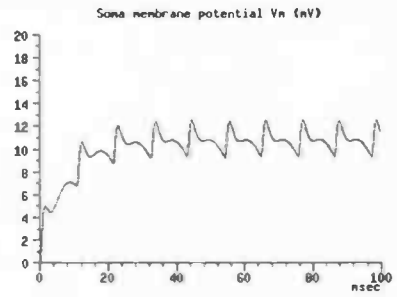

Figure 3.11: The resulting membrane potential in the soma when using a lower frequency than the preferred frequency, with a lower potential and less spikes.

frequency corresponding to the traveling time of the dendrite.

3.4.2 Higher Frequency

When the frequency is higher than the preference frequency, the interstimulus time is smaller. The expectation is, again, that the response is not optimal, because the response peaks do not coincide.
The pulses come faster, with the consequence of much overlay and integration. The activation is higher, only because the stimulation is faster. Many action potentials are generated. The effect that the response peaks do not match is lost.
Figure 3.12 shows the response of using a higher frequency than the preferred frequency.

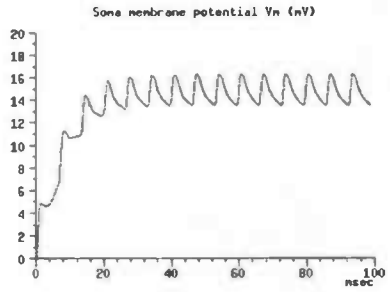

Figure 3.12: The resulting membrane potential in the soma when using a higher frequency than the preferred frequency, with a higher potential and more spikes.

3.4.3 Inhibition

Normally, excitatory inputs are not the only inputs that are supplied to the neuron. There exist also inhibitory inputs, who decrease the potential. The expectation is that an inhibitory input to the soma reduces the integration effects.
Used is only the pulse train of compartment 10. An inhibition input of $0.08e^{-5}$ mS is provided to the soma, at the same time as the excitatory input is provided to compartment 10.
The inhibition is designed like a K-channel, by modifying the alpha-function, Equation 1.17 of the synaptic input. The value for g_{max} is increased and the value for t_p is decreased, with the consequence that the inhibition starts later and has a longer duration.
The inhibition indeed resulted in sharper peaks, see Figure3.13. The inputs could integrate better. This is interesting, because this means input timing maybe is very exact.

The inhibition is also designed in a way that the excitatory input from compartment 10 and the inhibitory input from the soma together have the same base level, for each stimulus frequency. This means that more inhibition is provided to an input of a higher frequency (closer to the reversal potential) and less to a lower frequency. The excitatory input from compartment 1 only helps to trigger the potential above the threshold.
If the same input from compartment 1 is provided to the neuron for the different frequencies, the output of the higher frequency is still larger. This means that the overlay and integration effects are still too large. Consequently, inhibition did not solve the problem.

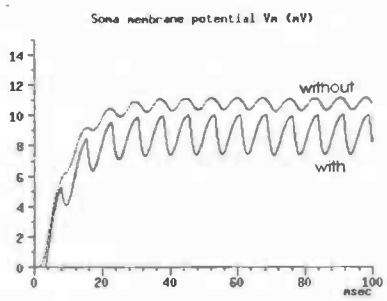

Figure 3.13: The resulting membrane potential in the soma when providing an inhibitory input to the soma, at the same time as an excitatory input is provided to compartment 10.

Chapter 4

Discussion

The influence of the low-pass property and the leaky integration behaviour of the neuron is large, while the neuron is very robust for variations in τ and λ. There is nothing to do about this, it is caused by the properties of passive neurons.
Changing τ with the consequence of less integration leads to neurons with a smaller or larger traveling time. The physiological parameter value ranges known from literature entail that the maximal length of the neuron is relatively small.
The frequency sensitivity cannot be improved with a longer dendrite, at least not with our approach.
The freedom to choose parameter values is quite large for most of the parameters, while for all that there is only little freedom to keep all the parameter values at the same time within the ranges of the literature of anatomical studies.
It would be nice if the idea had worked out, but this is also a result. It is an unexpected result, due to the temporal differences. After this work, the question still remains why the neuron has longer dendrites. Other studies can be done to investigate the precise timing of pulses in the soma, especially using inhibition. The bipolar property of the simulated neuron can still be used to perform some simulations.
Maybe interactions between neurons are important for the investigation of an improvement of frequency sensitivity in longer dendrites. Then a simulation of a neural network, with several neurons, is required to get results, instead of a simulation of only one neuron.

Neurons are fascinating. After this research I am sure about that. However, there are a few issues to find out before making this point.

I think that, to program efficiently, one first must know how the programming language works. This is difficult, because most of the time it seems better to start programming before the knowledge of the theory is good enough. This is also true for GENESIS. With the theoretical knowledge, it is far easier to program, and working with it is nicer.
Another issue is that I found out that it does not help to go on forever when it is unclear how to solve the problem you walk against. Neither does thinking about it day and night.

Summary

In Nature (vol 393, 1998, page 268-272), Agmon-Snir et al. showed that dendrites of coincidence detector cells in the auditory brain stem help to improve sound localization. They used short dendrites to reduce the effect of saturation at the dendrites in high-frequency coincidence detection cells. On the picture in the article one can see that they used an unrealistically short dendrite in their simulation. The question rises why the dendrites are longer in reality.
The coincidence detection cells are important because they combine frequency sensitive information from two ears. One might ask if this frequency sensitivity can be improved with longer dendrites. A large frequency sensibility improves the capability to ascertain the frequency and direction of the provided signal.

The idea is to use the traveling time of the dendrite to combine two inputs. This works for neurons with realistic traveling times for harmonics, with periods within 2-13 *msec*. The integration of input from the tip of the dendrite with input provided close to the soma might give a neuron sensitive to a particular period.
For this purpose, I simulated a neuron with a soma and two dendrites in GENESIS (Bower and Beeman, 1997). The dendrites were longer (and of a realistic electrotonic length) than the dendrites used by Agmon-Snir.
Before the simulation is designed, the behaviour of neurons is studied in more detail. Next, the simulated environment of GENESIS is studied and the model is designed.
To ascertain that the simulated neuron worked as expected and as a validation of the program, I performed some experiments.
Not much is known about the physiological parameters of this particular neuron. The parameter values of the simulated model should, therefore, preferably fall within the parameter ranges known from the literature. Deviations from this range should be justified.
By varying values of the time constant τ and the space constant λ, different

neurons can be designed. The preference period of the neuron is defined as the time an input needs to travel from the tip of the dendrite to the soma or close to the soma.
This period was used to design a corresponding pulse train that was used as input in the tip of the dendrite to a dendritic position close to the soma. These inputs should result in a regular pattern with a certain number of spikes in the axon. The responses of the two inputs arrive at the same time in the soma and give a potential with a maximal response if the peaks of the contributions coincide.
The wish was that with deviating input signals the output spike rate should be a decreasing as a function of the deviation with the preference frequency.

The leaky integration behaviour of the neuron posed as an unavoidable problem. The input from the end of the dendrite is completely spread when entering the soma. The outputs of the two peaks could not match, because there hardly was a peak in the response of the input at the end of the dendrite (altough there was a significant raise in the base level).
This posed a fundamental obstacle. We looked for ways to circumvent the obstacle. We thought that inhibition in the soma could make the shape of the output sharper so the inputs could integrate better. And indeed the inhibition resulted in a sharp peak. This was interesting because this means that input timing might be very precise. But with the use of inhibition the original idea of integration of two inputs was abandoned.

What we expected to be a positive point, namely that an interstimulus time of the same order as the traveling time of the dendrite, worked out to be adversely. Consequently, inputs with a stimulus frequency different from the preferred frequency react as if the stimulus frequency was the preferred frequency.
The question still remains what the purpose of longer dendrites is.

Bibliography

[1] Agmon-Snir, H., Carr, C.E. and Rinzel, J.: The role of dendrites in auditory coincidence detection. *Nature* **393**, 268-272 (21 May 1998)

[2] Arbib, M.A.: The handbook of Brain Theory and Neural Networks *Oxford University Press, Inc, New York* (1995)

[3] Bower, J.M. and Beeman, D.: The book of GENESIS, second edn. *Springer-Verlag New York, Inc* (1997)

[4] Hille, B.: Ionic Channels of Excitable Membranes, second edn. *Sinauer Associates Inc, Sunderland, MA* (1992)

[5] Koch, C.: Biophysics of Computation: Information Processing in Single Neurons. *Oxford University Press, Inc, New York* (1999)

[6] Koch, C. and Segev, I.: Methods in Neuronal Modeling: from synapses to Networks. *MIT Press, Cambridge, Massachusetts and London, England* (1989)

[7] Segev, I.: Sound grounds for computing dendrites. *Nature* **393**, 207-208 (21 May 1998)

[8] Weiss, T.F.: Cellular Biophysics, Volume 2: Electrical Properties. *MIT Press, Cambridge, Massachusetts and London, England* (1996)