

The modelling of neurons in the cochlear nucleus using Izhikevich neurons

Klaas Jacob de Vries
supervision: Ronald van Elburg

Abstract

An important function of our auditory system is its capability of determining the direction of an incoming sound. Several parts of the auditory system are involved in this task, one of which is the cochlear nucleus. If one were to build a complete model of the sound localization capabilities of the auditory system a model of the cochlear nucleus, and hence the neurons in the cochlear nucleus.

Current models for the cochlear nucleus neurons are variations of Hodgkin-Huxley conductance-based models. While being very accurate and biologically plausible they also rely on extensive intimate knowledge of the biophysical properties of the cell. This, combined with their computational costs, makes them impractical for constructing larger systems. A different model would be needed.

The model needed would have to be simple enough to be created without too much intimate knowledge about the biophysical properties of the cell, and preferably be computationally fast. Also it would have to be extensive enough to duplicate all possible behaviors of the cell. Standard leaky integrate-and-fire neuron models won't suffice. Although simple and fast enough, they are not capable of reproducing the complete behavior of biological neurons in the cochlear nucleus.

The Izhikevich model however claims to be a fast model for spiking neurons which is capable of demonstrating all desired behaviors of biological neurons. Also only very limited knowledge of the cell's biophysical properties are needed to create a model for this cell. This would make the Izhikevich model ideal for modelling neurons in the cochlear nucleus.

1 Introduction

It is hard to imagine a proper functioning intelligent being in a real-world environment without some form of sensory perception. Most prominent of the sensory systems, at least for humans, would be vision. There has been a lot of research to this aspect of perception, but it is of course not the only means we have to perceive our surroundings. A lot of information can also be gathered from the auditory system.

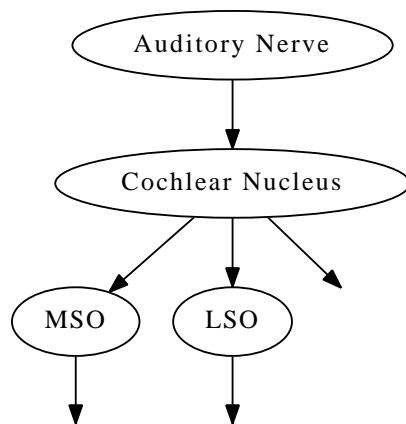


Figure 1: A very simplistic overview of the role of the Cochlear Nucleus in the auditory system. The first stage of audio processing occurs in the Auditory Nerve, which maps to the Cochlear Nucleus. The Cochlear Nucleus then maps onto the Medial- and Lateral Superior Olives (among others) where respectively the interaural time- and intensity differences are processed to determine the direction of the incoming sound.

1.1 Sound localization

One of the aspects of our auditory system is its ability to determine the direction of an incoming sound signal[2][12]. This can be done by two methods. One is by the interaural time difference[10] which works by the time difference, and hence phase difference, between the signal arriving at the left ear and the signal arriving at the right ear. The other is by the interaural intensity difference which works by the difference of intensity between the two sound signals at both ears. For low-frequency sounds the former method is used. For high-frequency sounds the latter method has to be used, due to the small wavelength compared to the distance between the ears, and because of the dampening effects of the skull.

In the auditory system, the first stage of processing an audio signal is done by the auditory nerve which connects the hair fibers in the ear to the cochlear nucleus (see figure 1). The cochlear nucleus in turn maps to other parts of the auditory system, including the medial superior olive where the interaural time difference is processed[1], and the lateral superior olive where interaural intensity difference is processed[18]. Thus the cochlear nucleus is the first step in the processing chain of localization information of an audio signal.

1.2 Modelling sound localization

If one were to build a model of the localization ability of the auditory system it is necessary to model the different parts of the auditory system involved in the toolchain of sound localization. Work has been done to model the interaural time difference processing in the medial superior olive [9]. For the auditory nerve models exist as well [17].

This research however focuses on the cochlear nucleus, and in particular the neurons in the cochlear nucleus. For this, models of individual neurons do exist [16], but they have downsides. This model is an implementation of the Hodgkin-Huxley model [5] in the Neuron software package. This model, while biologically accurate, may not be the best option for use in a more complete model of the auditory system. Besides being relatively slow, it also suffers from the downside that a lot of experimental data is needed to construct a single neuron, which would be impractical to do for larger systems containing lots of neurons.

A much simpler solution would be the use of leaky integrate-and-fire neurons. While simpler, they are however not sufficient to capture all possible behaviors of the neurons in the cochlear nucleus, as described by Evans [3]. It can for example not display bistability of phasic spiking [7].

Another type of neuron model is the Izhikevich model [8]. This remarkably simple model has the advantage that only very few physiological aspects of a neuron are needed to model the neuron by fitting it using experimental data. Also it claims not to suffer from the limitations of the leaky integrate-and-fire neurons [6].

In this research an attempt will be made to use the Izhikevich neuron model to simulate the neurons found in the cochlear nucleus.

2 Background

2.1 Types of neurons in the cochlear nucleus

In research done by Evans [3] several different types of neurons in the cochlear nucleus are identified (see figure 2). Each of these types have their own pattern of excitatory or inhibitory response to sound stimuli of different frequencies and intensities. As information about the frequency or intensity of a stimulus is needed to determine the interaural time difference or interaural intensity difference respectively this is an important step in determining the direction of the stimulus.

Several implementations for models in the cochlear nucleus can be found on the model database for the Neuron modelling software (<http://senselab.med.yale.edu/modeldb/>). These models are based on the Hodgkin-Huxley model. One of the models, which is used in this research, is the model created by Rothman and Manis [15]. This model is an implementation of the octopus-cells, or in the terminology of Evans, the 'On'-cell.

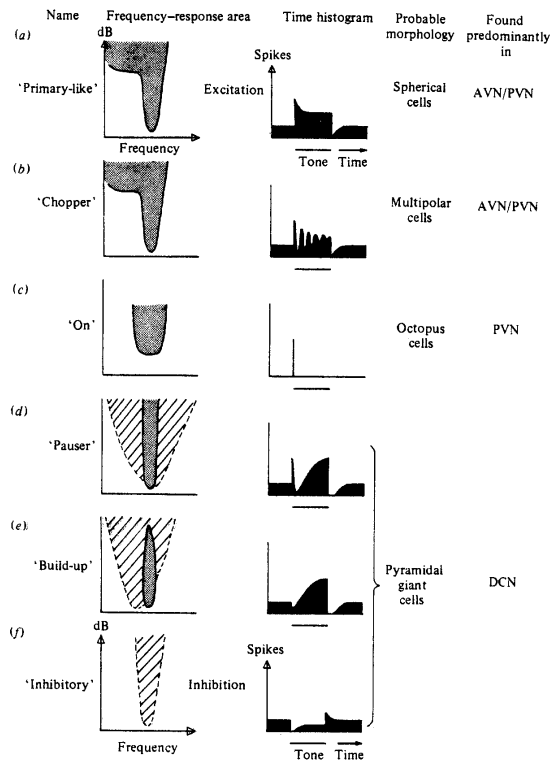


Figure 2: The types of neuron cells found in the cochlear nucleus, as identified by Evans. The images in the left-hand column show the response of the neuron type depending on the frequency and intensity of the stimulus, which can be excitatory (stippled) or inhibitory (shaded). The images in the right-hand column show the number of spikes related to the tone used for input. Next to the left-hand images are the nicknames Evan's gave to the several types of neurons he identified. Image taken from [3].

2.2 Neuron models

2.2.1 Leaky integrate-and-fire

The leaky integrate-and-fire neuron is a very simple and computationally efficient model for a spiking neuron. Probably because of it being simple and fast it is also one of the most widely used models. The equation for the leaky integrate-and-fire model is:

$$v' = I + a - bv, \text{ if } v \geq v_{\text{thres}}, \text{ then } v \leftarrow c \quad (1)$$

Where v is the membrane potential, I is the input current and a , b , c and v_{thres} are parameters. When a sufficiently large input current is given the membrane potential will increase until it reaches v_{thres} , it's threshold value. At this time a spike is fired and the membrane potential is reset to v .

As the only variable in this model is the membrane potential v this model is not sufficient to display all characteristics found in biological cells. For example bistability or rebound responses [7]. This makes this model unsuitable for our purpose of modelling the neurons in the cochlear nucleus.

2.2.2 The Hodgkin-Huxley model

The Hodgkin-Huxley model [5] is a far more complex model than the leaky integrate-and-fire model. In its essence it is a description of the separate Na and K ion-channels in a neuron. For this model the current through each ion-channel, and hence the conductance of each ion-channel, is a parameter that has to be determined to create a model. This has the advantage that each parameter has a clearly identifiable physical meaning, which makes this type of model of great use for biological research.

However, having to determine all the conductances of all ion-channels, this means a lot of time-consuming measurements would have to be done to model a given neuron in the cochlear nucleus. Another disadvantage of this model is the computation needed. Where as the leaky integrate-and-fire model would only need a handful of floating-point operations at each time step the Hodgkin-Huxley model would need an approximate 120 floating point operations [7].

2.2.3 The Izhikevich model

The leaky integrate-and-fire being too simple to display all desired characteristics and the Hodgkin-Huxley model being too slow and impractical for use in larger systems one would like an alternative. Izhikevich claims to have such a model presented in [6] and [8]. While slightly more complex than the leaky integrate-and-fire model, his comparison of different types of spiking neuron models shows it to be still very computationally efficient [7]. At the same time the claim is also made that the model is biologically plausible enough to display all characteristics needed for modelling cortical neurons. (It should be noted however that Izhikevich takes biologically plausible to mean that the model is capable of demonstrating all

```

C=100; vr=-60; vt=-40; k=0.7;           % parameters used for RS
a=0.03; b=-2; c=-50; d=100;           % neocortical pyramidal neurons
vpeak=35;                               % spike cutoff

T=1000; tau=1;                           % time span and step (ms)
n=round(T/tau);                          % number of simulation steps
v=vr*ones(1,n); u=0*v;                  % initial values
I=[zeros(1,0.1*n),70*ones(1,0.9*n)];% pulse of input DC current

for i=1:n-1                               % forward Euler method
    v(i+1)=v(i)+tau*(k*(v(i)-vr)*(v(i)-vt)-u(i)+I(i))/C;
    u(i+1)=u(i)+tau*a*(b*(v(i)-vr)-u(i));
    if v(i+1)>=vpeak                       % a spike is fired!
        v(i)=vpeak;                       % padding the spike amplitude
        v(i+1)=c;                         % membrane voltage reset
        u(i+1)=u(i)+d;                     % recovery variable update
    end;
end;
plot(tau*(1:n), v);                       % plot the result

```

Figure 3: Matlab-code for the Izhikevich model for spiking neurons in all it’s glory. Taken from [8].

the characteristics of a biological neuron, not that the model is an accurate description of a biological neuron where all or most parameters have a clear physical meaning.)

The Izhikevich model is described by two differential equations.

$$C\dot{v} = k(v - v_r)(v - v_t) - u + I \quad (2)$$

$$\dot{u} = a\{b(v - v_r) - u\}, \text{ if } v \geq v_{\text{peak}}, \text{ then } v \leftarrow c, u \leftarrow u + d \quad (3)$$

In these equations k is the input resistance v_r the resting membrane potential, v_t the threshold potential, C is the cell’s capacity and I is the input current. The parameters a , b , c and d are dimensionless parameters.

To demonstrate the relative simplicity of this model the Matlab source code for this model, in all it’s glory, is listed in figure 3.

For this model four parameters, being k , v_r , v_t and C , would have to be determined by biological data about the to modelling cell. The a , b , c and d parameter could be fit to the experimental data about the input-output relation of the cell. Note that for this it is not necessary to examine all the inner details about the cell, just the input-output relation.

3 Methods

3.1 Software systems

Our little neuron is not meant to live by itself. The ultimate purpose of it would be to act in a complete system for sound localization. Therefore it would be best to place it among existing models. Best would be to make the model in such a way that it integrates well with existing models of the

auditory nerve and possibly the medial- and lateral superior olives. Of these especially integration with an auditory nerve model would be useful as it would make the model capable of directly responding to arbitrary real-world sounds as opposed to abstract input currents.

Models of the auditory nerve exist which are implemented in the dsam auditory modelling software [11]. The dsam software has already been successfully combined with the csim neural circuit simulator [4]. The csim software package also has an implementation of the Izhikevic neuron model. This would make the csim package ideal for usage in the experiments.

However if one is to fit the model one needs experimental data to fit it with. As 'experimental data' the model of an octopus cell by Rothman and Manis is used (see section 3.2.1). This existing model is implemented in the Neuron software package and is a model of the neuron cell only. It is not connected to the auditory nerve or even a synapse and uses a current clamp for it's input.

If one where to match the output of the Rothman and Manis model against the Izhikevich model given an arbitrary input sound one would also need to equip the Rothman and Manis model with an equal or equivalent auditory nerve model and synapse model. After all: the neuron model needs to be fitted, not the synapses. Alternatively one could take the easier approach of feeding the current clamp directly to the Izhikevich model instead of an arbitrary sound to the auditory nerve model. For now the easier approach of feeding the current clamp directly to the Izhikevich model is chosen.

Once a current clamp, as opposed to a sound, is used as input for the Izhikevic model the need to use the dsam auditory nerve, and hence the use of the csim Izhikevich implementation, disappears. The simple Matlab-implementation by Izhikevich himself can be used [8](listed in figure 3), with some small adaptations to allow for an arbitrary current clamp.

3.2 Data Fitting

The construction of the Izhikevich neuron model for a cochlear nucleus neuron ultimately boils down to fitting the four non-physical parameters of the Izhikevich model. The four physical parameters: the input resistance k , the resting membrane potential v_r , the threshold membrane potential v_t and the cell's capacity C are taken from the existing model by Rothman and Manis described in the next section.

3.2.1 Data

For data fitting one needs data. To model the behavior of a neuron one needs to know how the neuron is supposed to behave. Ideally one would like some actual experimental data. In this research for practical reasons an existing model is chosen. It's input is duplicated and it's output is recorded. This output is then for all practical purposes regarded as experimental data. As the internal details of the model are ignored and only the input and output is considered the techniques used would work equally well on true experimental data.

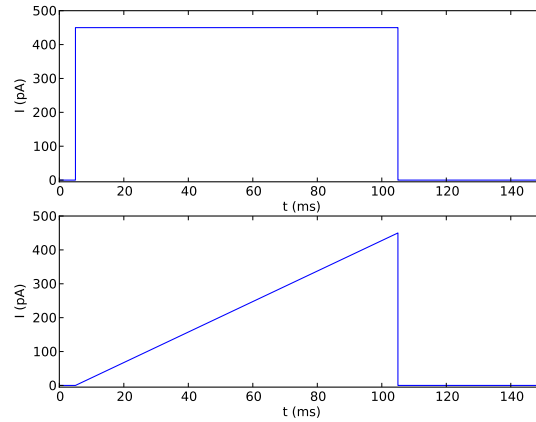


Figure 4: The input stimuli used for fitting. A block stimulus (top) and a ramp stimulus (bottom). Note that the stimuli do not stop immediately after the input current is gone, but a '0-signal' is continued for another 45 ms.

For their researches ([15], [13], [14]) Rothman and Manis have developed a model of the octopus cell in the cochlear nucleus. This model is implemented using the Neuron modelling software. As a stimulus a current clamp is used which can be duplicated. During running the model the cell's membrane potential over time is recorded.

As the cell's membrane potential is continues data. Ultimately however we want to model the cell's behavior and not every little fluctuation in it's membrane potential. That is: we want to match the cell's spike trains. For this the membrane potential data needs to be converted to the matching spike train. This is done by recording all time indices where the membrane potential crosses a given threshold value.

For the input stimuli a block stimulus and a ramp stimulus is provided (see figure 4). Both the stimuli have an onset delay of 5 ms, a duration of 100 ms. After the stimulus ends a current of 0 A is presented for another 45 ms. This is important because not only the starting of a spike train but also it's stopping needs to be modelled.

3.2.2 Genetic algorithm

The eventual data fitting can be viewed as a function optimization problem. Given an input (current clamp) and a reference output, how can we tune the parameters of the model such that the difference between it's output and the reference output is minimal. This difference is to be determined by an error function (or fitness function), see section 3.2.3 for details.

There exist many different approaches for optimization problems. Here a genetic algorithm is chosen. This because by more direct approaches like gradient descent there is a high probability of getting stuck in a local

minimum. The implementation of the genetic algorithm used is the one present in Matlab's global optimization toolbox.

The initial conditions for the four parameters to be fit are derived from figure 8.8 in Izhikevich book [8]. In this figure several different types of spiking neurons are displayed and their corresponding values for the a , b , c and d parameters can be found in the source code published on the website www.izhikevich.org. By eyeballing the graphs the initial conditions for the parameters are chosen.

Once the genetic algorithm is finished it's solution is fine-tuned by using it as the initial value for a gradient descent algorithm. The implementation for this gradient descent is the `fminsearch` function in Matlab.

3.2.3 Fitness function

For a genetic algorithm to work, or for that matter any function minimization algorithm, a function is needed. This fitness function, or error function, has to be constructed in such a way that it yields a high value for a bad fit and a low value for a good fit and is more or less continuous.

In our context it would be natural to take the membrane potential over time of the Rothman and Manis reference model and the Izhikevich model. The time integral of the squared difference between both would be the return value of the error function. However, because we want to model the cell's behavior and hence it's spike pattern as opposed to the membrane potential this is not really suitable. The spikes can be seen in the membrane potential but because they are, though very high, also very thin. This means they have only have a small contribution to the time integral and there would be hardly any penalty for missing the spikes. In fact a fit which would yield a flat line at the cell's resting potential would only yield a small error but would model the cell's behavior at all!

We would like to use the actual spike pattern in our fitness function. One problem with this is that a spike pattern is not in any way a continuous function. Also we would like our function to have the following characteristics. It should yield a high error if a spike is missed. It should yield an error if a spike is not fired at the correct time. On the other hand, it should not be judged to harshly if a spike is missed in some distant past, i.e. if all spikes but one match perfectly we are reasonably happy.

To achieve this the following function is defined for both spike patterns on each time index t :

$$f(t) = \sum_{t_s < t} e^{-d(t-t_s)} \quad (4)$$

Where the t_s denote the time indices of the spikes and d is a small but positive decay parameter. This value should not be chosen to large as the penalty for missing spikes would become to small. Also it should not be chosen to large as the penalty for not firing at the right time would become to small. Finally, as this is a exponentially decaying function the error of a missed spike won't 'remain in memory' forever. The decay parameter d is chosen by eyeballing the graphs like those in figure 5. Note that equation 4 is technically not a continuous function, but it will do for our

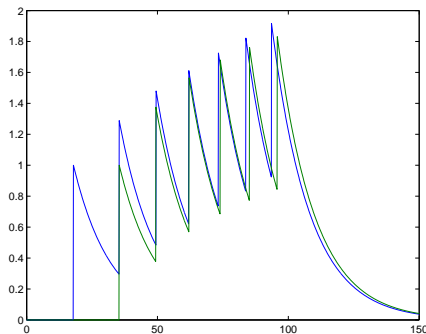


Figure 5: Visualization of the error functions. The blue line is error function of the reference spike pattern. The green line is the error function of the spike pattern of model. Each peak indicates a spike being fired after which it slowly decays. The area which makes up the difference of the two lines would be the error. Note that this achieves all of yielding a big penalty of missing a recent spike, a little penalty for missing a not so recent spike and a penalty for not having a spike at the right time.

current purposes. Now our error function can be defined as:

$$e = \int_t (f_1(t) - f_2(t))^2 dt \quad (5)$$

Where f_1 is the function f , as in equation 4, corresponding to the data and f_2 the same function corresponding to the output of the model. This error function achieves all of the aforementioned characteristics. A visualization of this can be seen in figure 5.

4 Results

4.1 Block stimulus

The Izhikevich model is fitted using a current clamp in the form of a block stimulus as shown in the top image of figure 4. Some physical parameters are taken from the existing Rothman and Manis model of the cochlear nucleus cell, being the input resistance k , the resting potential v_r , the threshold potential v_t and the cell's capacitance C . Their values can be seen in table 1.

The genetic algorithm is then used to find the four non-physical dimensionless parameters for the model, being the a , b , c and d parameters. The values found can also be found in table 1.

In figure 6 a comparison between the 'experimental data' from the Rothman and Manis model and the fitted Izhikevich model can be seen. The spike pattern for the Rothman and Manis model is in the middle

Table 1: The parameters for the Izhikevich model for the block stimulus.

k	0.7
v_r (mV)	-63.9
v_t (mV)	-40.0
C (pF)	12.0
a	0.716742
b	23.155029
c	37.622159
d	374.404241

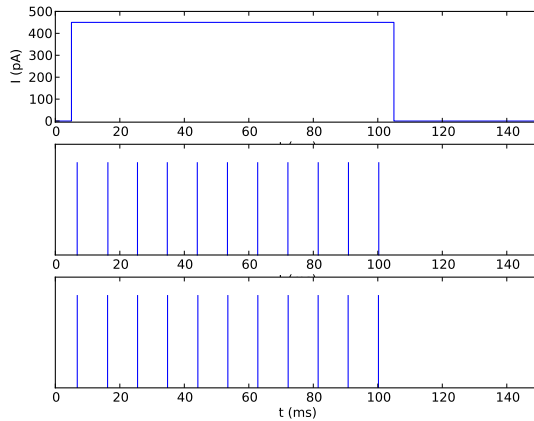


Figure 6: Graphs of the result of the model when fitted to a current block. The top image shows the input current clamp. The middle image shows the spikes fired by the existing model by Rothman and Manis. The bottom image shows the result of the Izhikevic model. As can be seen the Izhikevich model is a near-exact match with the existing model.

Table 2: The parameters for the Izhikevich model for the ramp stimulus. Note that only the a , b , c and d values differ from those in table 1.

k	0.7
v_r (mV)	-63.9
v_t (mV)	-40.0
C (pF)	12.0
a	0.007580
b	-1.229707
c	-162.805298
d	67.928863

image and the pattern for the Izhikevich model is in the bottom image. It should be clear that they are near-identical. They both have the same number of spikes and the spikes are being fired at the same time. Moreover in both case spikes are no longer being fired after the input current disappears.

4.2 Ramp stimulus

The Izhikevich model is also fitted on the current clamp in the form of a ramp stimulus like the one in the bottom image of figure 4. As only the non-physical dimensionless parameters are fitted the other parameters are equal to the ones used for the block stimulus.

Again the genetic algorithm is run to find suitable values for the a , b , c and d parameters. The results can be seen in table 2.

The comparison between the 'experimental data' and the Izhikevich model can be seen in figure 7. All spikes from the data save one are reproduced by the model. It should be noted that their timing is not exact and the Izhikevich model completely misses the first spike.

4.3 Block and ramp stimulus

Finally the model is fitted using the block and ramp stimuli simultaneously. This is done by evaluating the error function of equation 5 for both the block spike pattern and the ramp spike pattern and adding them up. The found parameters can be seen in table 3.

Once the model is fitted on both the block and ramp stimulus it can be tested against the block or ramp stimulus. In figure 8 the comparison with experimental data for the block stimulus is shown. Only two spikes match.

The comparison between the Izhikevich model and the experimental data for the ramp stimulus can be seen in figure 9. Here one spike is matched at 35 ms. Two spikes are fired in rapid succession by the Izhikevich model at 82 ms. Both of these can be said to match the spike in the experimental data with explicitly noting that the experimental data only has one spike at this time.

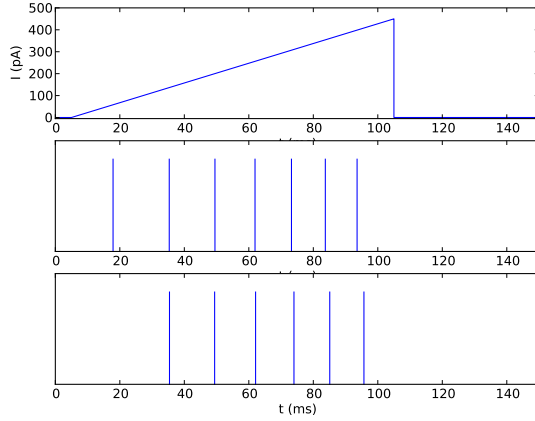


Figure 7: Graphs of the result of the model when fitted to a current ramp. The top image shows the input current clamp. The middle image shows the spikes fired by the existing model by Rothman and Manis. The bottom image shows the result of the Izhikevich model. It can be seen that the Izhikevich model is capable of reproducing all spikes save the first.

Table 3: The parameters for the Izhikevich model fitted on both the block and the ramp stimulus simultaneously

k	0.7
v_r (mV)	-63.9
v_t (mV)	-40.0
C (pF)	12.0
a	-0.117710
b	0.006190
c	0.480803
d	6.892889

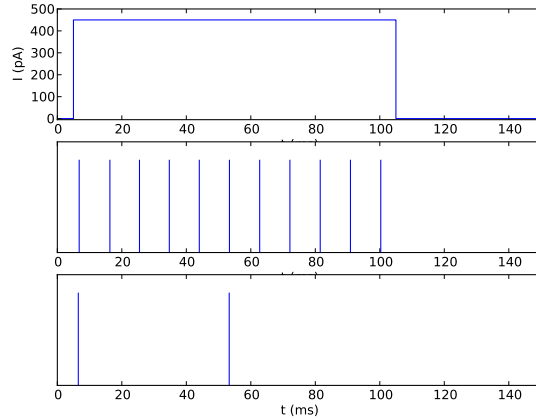


Figure 8: Graphs of the result of the model when fitted to both the current and the block stimuli and tested against the block stimulus. Again, the top image is the input current, the middle image is the spike pattern of experimental data and the bottom image is the spike pattern of the Izhikevich model. Only two spikes are matched.

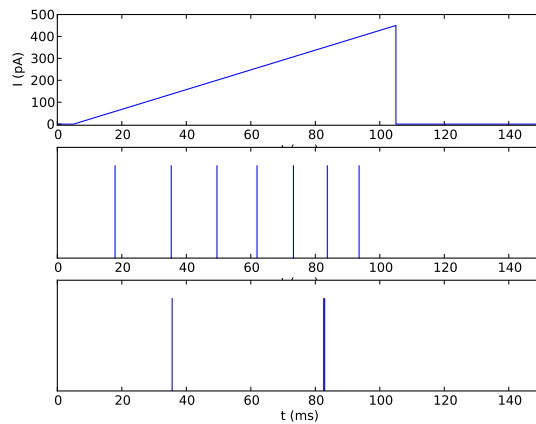


Figure 9: Graphs of the result of the model when fitted to both the current and the block stimuli and tested against the ramp stimulus. As before the top image is the input current, the middle image is the spike pattern of experimental data and the bottom image is the spike pattern of the Izhikevich model. One spike at 35 ms is matched. The Izhikevich model fires two spikes in rapid succession at 82 ms, both of which can be considered to match with the experimental data. None of the other spikes are matched.

5 Discussion

5.1 Matching the stimuli

5.1.1 Block stimulus

The approach of using the genetic algorithm to fit the Izhikevich model works remarkably well when using the block stimulus. As can be seen in figure 6 the behavior of the Izhikevich model is nearly identical to the behavior of the existing Rothman and Manis model. The differences in the timing of the spikes are very small. Because the Rothman and Manis model runs on a higher time resolution than the Izhikevich model used this difference in time can be attributed to this difference in time resolution and can be neglected. As the Izhikevich model makes no use of the internal details of the Rothman and Manis model it can only be assumed that the same results will be obtained when fitting on real experimental data.

5.1.2 Ramp stimulus

Also on the ramp stimulus the fitting of the Izhikevich model works almost as desired. The fit as can be seen in figure 7 is however not as clean as it is with the block stimulus. One spike is missed, and some other spikes show differences in the timing. Though these differences are small they are too large to be simply be the result of different time resolutions. Nevertheless the behavior of the Izhikevich model is nearly identical to the existing Rothman and Manis model. So far this also looks promising for the Izhikevich model and this approach of fitting its parameters.

5.1.3 Block and ramp stimulus

However one disturbing fact arises. When comparing the parameters of the block stimulus found in table 1 with the parameters of the ramp stimulus in table 2 it shows that they are very different. To model the behavior of a cell it is not enough to just model the behavior *given an input*, but the behavior *given any input* should be modelled.

Therefore a good model for a neuron should not only give a good match with the experimental data for the block stimulus or the ramp stimulus. It should also give a good match when fitted on both the block and ramp stimulus. Sadly as can be seen in figures 8 and 9 results are not so encouraging. The genetic algorithm fails to find a solution which yields the correct behavior for both the block and ramp stimulus and ends up with a solution which yields correct behavior on neither of them.

The fundamental question in this issue is who is to blame? Is this perhaps a limitation of the Izhikevich model which perhaps is capable of demonstrating nearly every possible behaviors biological neurons but not with the same set of parameters. Or is this a limitation to the approach of trying to find these parameters using a genetic algorithm. Does there exist a combination of parameters which does reproduce all behaviors of the cell but is unlikely to be found by these means?

5.2 Limitations of scope

This research has only used a limited set of stimuli. Even if both the block and ramp stimulus could be matched simultaneously only a limited set of input's a real-world neuron would encounter would be accounted for. A chopper pattern would be a good next step.

Ultimately one would like to take it to a higher abstraction level however. One would like to match the firing pattern of the cell not to an abstract input current clamp but to actual sound. One would like to create a model capable of reproducing and identifying the cell types as by the taxonomy by Evans [3], as seen in figure 2. For this one would need to connect the model to a model of the auditory nerve capable of processing arbitrary sound, or at least sounds of an arbitrary frequency and intensity. Also a model for the synapses for the cochlear nucleus would be needed.

If these models of the auditory nerve and the synapse would exist in a suitable manner they could be connected to Izhikevich model as well as to the Rothman and Manis model. This way the Rothman and Manis model could serve as 'experimental data' the same way it did in this research.

Another limitation of this research is that only one of the types of neurons in the cochlear nucleus is examined. There are however no good reasons not to assume that the same techniques could be used to model the other types of neurons in the cochlear nucleus. A prerequisite for this would be of course the availability of existing models or experimental data on these types of neurons.

6 Conclusion

The Izhikevich model is very capable of simulating the neurons in the cochlear nucleus when tested on a block stimulus. To a lesser extent this is also true when tested on a ramp stimulus. This is promising as the Izhikevich model is computationally much faster than existing conductance-based models. Moreover this model does not require extensive knowledge of the physiological aspects of the cell.

However when the Izhikevich model has to reproduce the behavior of a neuron on different stimuli given the same set of parameters the approach of finding the parameters using a genetic algorithm breaks down. The parameters found no longer enable the Izhikevich model to give the correct behavior on all input stimuli.

The important and as yet unanswered question is whether this is a fundamental limitation of the Izhikevich model or whether the correct set of parameters does exist but is unlikely to be found using these methods.

References

- [1] A. Brughera, E. Stutman, L. Carney, and H. Colburn. A model with excitation and inhibition for cells in the medial superior olive. *Audit Neurosci*, 2:219–233, 1996.
- [2] J. Christensen-Dalsgaard and C. Carr. Evolution of a sensory novelty:

- tympanic ears and the associated neural processing. *Brain Research Bulletin*, 75(2-4):365–370, 2008.
- [3] E. Evans. Auditory processing of complex sounds: an overview. *Philosophical Transactions: Biological Sciences*, 336(1278):295–306, 1992.
- [4] B. Fontaine and H. Peremans. Tuning bat LSO neurons to interaural intensity differences through spike-timing dependent plasticity. *Biological Cybernetics*, 97(4):261–267, 2007.
- [5] A. Hodgkin and A. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500, 1952.
- [6] E. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on neural networks*, 14(6):1569–1572, 2003.
- [7] E. Izhikevich. Which model to use for cortical spiking neurons? *IEEE transactions on neural networks*, 15(5):1063–1070, 2004.
- [8] E. Izhikevich. *Dynamical systems in neuroscience: The geometry of excitability and bursting*. The MIT press, 2007.
- [9] Kruijne, W. Onderzoek naar modellen voor Interaural Time Difference detectie in de Medial Superior Olive in zoogdieren. 2009.
- [10] D. McAlpine, D. Jiang, and A. Palmer. A neural code for low-frequency sound localization in mammals. *nature neuroscience*, 4(4):396–401, 2001.
- [11] R. Meddis and L. OMard. A computer model of the auditory-nerve response to forward-masking stimuli. *The Journal of the Acoustical Society of America*, 117:3787, 2005.
- [12] R. Munkong and B. Juang. Auditory perception and cognition. *IEEE Signal Processing Magazine*, 25(3):98–117, 2008.
- [13] J. Rothman and P. Manis. Differential expression of three distinct potassium currents in the ventral cochlear nucleus. *Journal of neurophysiology*, 89(6):3070, 2003.
- [14] J. Rothman and P. Manis. Kinetic analyses of three distinct potassium conductances in ventral cochlear nucleus neurons. *Journal of neurophysiology*, 89(6):3083, 2003.
- [15] J. Rothman and P. Manis. The roles potassium currents play in regulating the electrical activity of ventral cochlear nucleus neurons. *Journal of neurophysiology*, 89(6):3097, 2003.
- [16] J. Rothman, E. Young, and P. Manis. Convergence of auditory nerve fibers onto bushy cells in the ventral cochlear nucleus: implications of a computational model. *Journal of neurophysiology*, 70(6):2562, 1993.

- [17] C. Sumner, E. Lopez-Poveda, L. OMard, and R. Meddis. A revised model of the inner-hair cell and auditory-nerve complex. *The Journal of the Acoustical Society of America*, 111:2178, 2002.
- [18] D. Tollin. The lateral superior olive: a functional role in sound source localization. *The neuroscientist*, 9(2):127, 2003.