August 2018

UR EAR Version 2.0 - User Manual

(Edited from UR EAR Version 1.0 User Manual by Afagh Farhadi & Laurel Carney)

A guide to our software tool: “University of Rochester: Envisioning Auditory Responses” (UR_EAR Version 2.0). For modeling, creating stimuli, and more. This modeling tool is upgraded version from the previous model UR_EAR_v1_0.

Table of Contents

Chapter 1 - Exploring the UR_EAR Package

This MATLAB code package is designed to run auditory models with different stimuli. The package contains numerous files and folders. The list is as follows:

Stimulus functions (see below)

Model_IHC_BEZ2018 (mex file versions for both PC and Mac)

model_Synapse_BEZ2018 (“” “”)

model_IHC (“” “”)

model_Synapse (“” “”)

ffGn

fitaudiogram2.m

generate_neurogram_UREAR2.m

generateANpopulation.m

SFIE

EI_KF_CD.m

get_alpha_norm

play.jpg

unitgain_bpFilter

UR_EAR Version1.0 User Manual

UR_EAR Version1.0 ReadMe

UR_EAR_v2_0

These files and folders are all necessary to make the modeling application run properly. The files the user is most likely to overtly use are the stimulus functions, and the main UR_EAR_V#_# m-file.

Stimulus functions

Each stimulus type is created by an m-file containing a function that is called by UR_EAR. The parameters are described and assigned to specific variable names to be able to interact with the rest of the UR_EAR code. The most general form of a stimulus, with only the overall sound level as a variable, is a *.wav file. For more information on how to add your own stimulus along with controllable parameters, see the Creating Your Own Stimulus section in Chapter 3.

Chapter 2 - Running and Using the GUI

Running and Using the GUI - the basics:

Once you have downloaded the package, you are ready to open up MATLAB and run the .m file called UR_EAR_2. Press “run”, and a GUI will open:

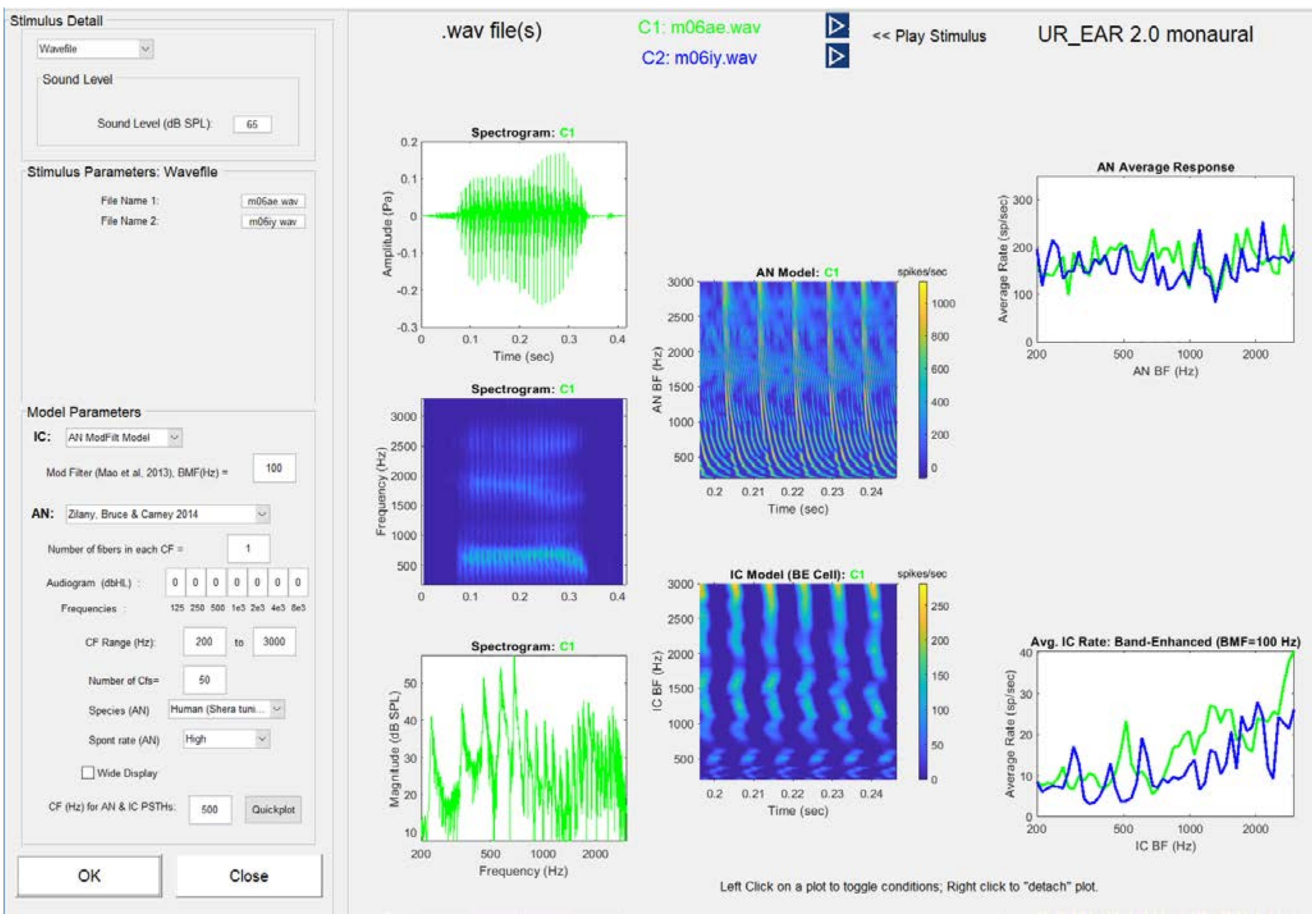
This GUI allows you to choose from a variety of models, stimuli, and parameter values, then see the resulting plots.

Stimulus Details:

At the top left is the “Stimulus Details” section, where you can select which stimulus you’d like to run from a pop-up menu. You can also specify stimulus and model details, such as sound level. Once a stimulus is selected, the “Stimulus Parameters” section at the bottom right will change based on your selection.

Model Parameters:

At the bottom left is the “Model Parameters” section, where you select which auditory nerve model and which IC model you’d like to use from the pop-up menus and specify the model parameters. Parameter selection will depend on which model you have selected, but generally there will be options such as auditory-nerve (AN) center-frequency range, number of AN fibers (evenly log spaced across the designated frequency range) and hearing status (by audiogram, a new feature in UR_EAR_v2_0). In UR_EAR_v2_0 there are two different AN models and three different IC models to choose from. The plots in the center column show a small portion (50 msec) of each response. You also have the option to examine the entire response using the “wide view” of the response plots by clicking on the related box. Another feature that is added to the GUI is the possibility of choosing the number of fibers in each CF; this feature allows averaging across a set of independent AN simulations at each CF.



Stimulus Parameters:

At the bottom right is the “Stimulus Parameters” section, where you can change the values of the parameters specific to the stimulus you selected in the “Stimulus Details” section. You can change the values of duration, frequencies of spectral edges or bandwidths, chose between various comparisons, etc. For some stimuli, there is an automatic comparison between two versions of the same type of stimulus, for example, comparing noise-plus-tone to the same noise without a tone. These plots are made at the same time, and plotted on the same chart. For some comparisons you will have to toggle between the different plots once you press “ok” and run the model. To compare various parameter changes that are not included automatically, simply run the model twice with those different selections. Due to its ease and speed, exploring the parameter space is easy.

Press “Ok”:

Once you have specified the desired values, press the “Ok” button to run the model. If running correctly, you should see evidence in the command line of MATLAB, such as the phrase “running…”. Otherwise, you will be able to see which errors occurred in the command line. After the model has finished running, the left panel will show various plots, toggling capabilities, and ‘play’ buttons to hear the stimuli you selected.

The Built-In AN and IC Models:

This software includes three IC models and two different AN model. The purpose of this code is to easily explore a variety of stimuli with various models, and having these built-in models contributes to that goal. These models can be used with both built-in stimuli and novel stimuli added by users. The three IC models are the Same-Frequency Inhibition and Excitation Model (Nelson & Carney 2004), the simpler Inferior Colliculus (IC) Modulation Filter Model (Mao et al 2013) and Coincidence-Detector Neural Cells Model (Krips & Furst 2009). The built-in AN models are Zilany, Bruce, & Carney 2014 Auditory Nerve (AN) Model and Bruce, Erfani, & Zilany 2018 Auditory Nerve (AN) Model. All of these models have various parameters that can be easily adjusted in our software. See Chapter 3 for more information on the sections of UR_EAR.m which run the models, plot the responses, and interact with parameter selection.

The Built-In Stimuli:

This software includes many built-in stimuli ready to interact with the built-in models. The stimuli are complex sounds based on psychophysical studies and built specifically for use with the built-in models. Our models focus on the inferior colliculus, and so the complex sound stimuli included were also intended for studying the inferior colliculus. Following is a list of default stimuli and their descriptions:

Wavefile:

We have included a wavefile option for type of stimulus. This is one of the simplest ways to interact with the GUI, although altering parameters each time the stimulus is run is less easy. The program will take any wavefile and give the response. We have included a few example wavefiles as the default vowel-consonant-vowels, ‘m06ae.wav’ and ‘m06iy.wav’.

Noise Band:

Gaussian noise stimulus, with a band specified by selecting the low and high frequency limits. This stimulus is useful for exploring *Edge Pitch,* based on Klein and Hartmann (1981). A pitch is perceived at about 4% inside the noise band from a sharp spectral edge. The IC population response near the ‘edge’ is intriguing… (compare it to the response to a tone plus noise.)

Notched Noise:

Notched noise is a commonly used masking stimulus for psychophysical experiments, such as in Patterson, 1976. The notch is centered at the signal frequency, and is used to derive shapes of critical bands.

Tone in Noise:

Another popular research topic in auditory processing, tone-in-noise paradigms involve discerning the difference between a stimulus of just noise, and of the same noise with an added tone.

Profile Analysis:

A stimulus waveform constructed of evenly log-spaced components centered around 1,000 kHz. This stimulus can be presented with and without an increment of the center frequency component. See Green (1983) for details.

Pinna Cues:

Filters band-limited noises using an artificial notch, modeling head-related transfer functions that simulates a simple spectral cue for sound localization.

SAM Tone (or pure tone):

Creates an AM tone with specified carrier and modulation frequency and depth. Set modulation depth to 0 for a pure tone.

Complex Tone / Harmonic Complex:

Specify the pitch and range of harmonics. Specify whether or not to include the fundamental.

Single Formant:

Simple triangular spectral peak, similar to those seen in vowels, a la Lyzenga & Horst 1995.

Double Formant:

Two formant peaks, generated using Klatt synthesizer. Similar to stimuli in Lyzenga & Horst 1997, 1998.

***Schroeder Phase Harmonic Complex*:**

Set F0 and # of components. Positive and Negative C values are compared.

Noise in Notched Noise:

A band of noise (target noise) is added to a complementary band-reject noise (masker noise) which is 10 dB higher in spectrum level. In test stimulus the target noise spectrum level is increased and in the standard stimulus the target noise level is fixed. Discrimination between these two stimuli was studied in Viemeister 1983.

FM tone:

Two stimulus both Sinusoidal frequency modulated Tone with two different modulation frequency and modulation excursion that you can adjust them in GUI.

Forward Masking:

First condition is a sinusoid masker with a short delay before the probe tone, and second condition is just the masker. You can change properties of both masker and tone in stimulus parameters in GUI.

***CMR-BW*:**

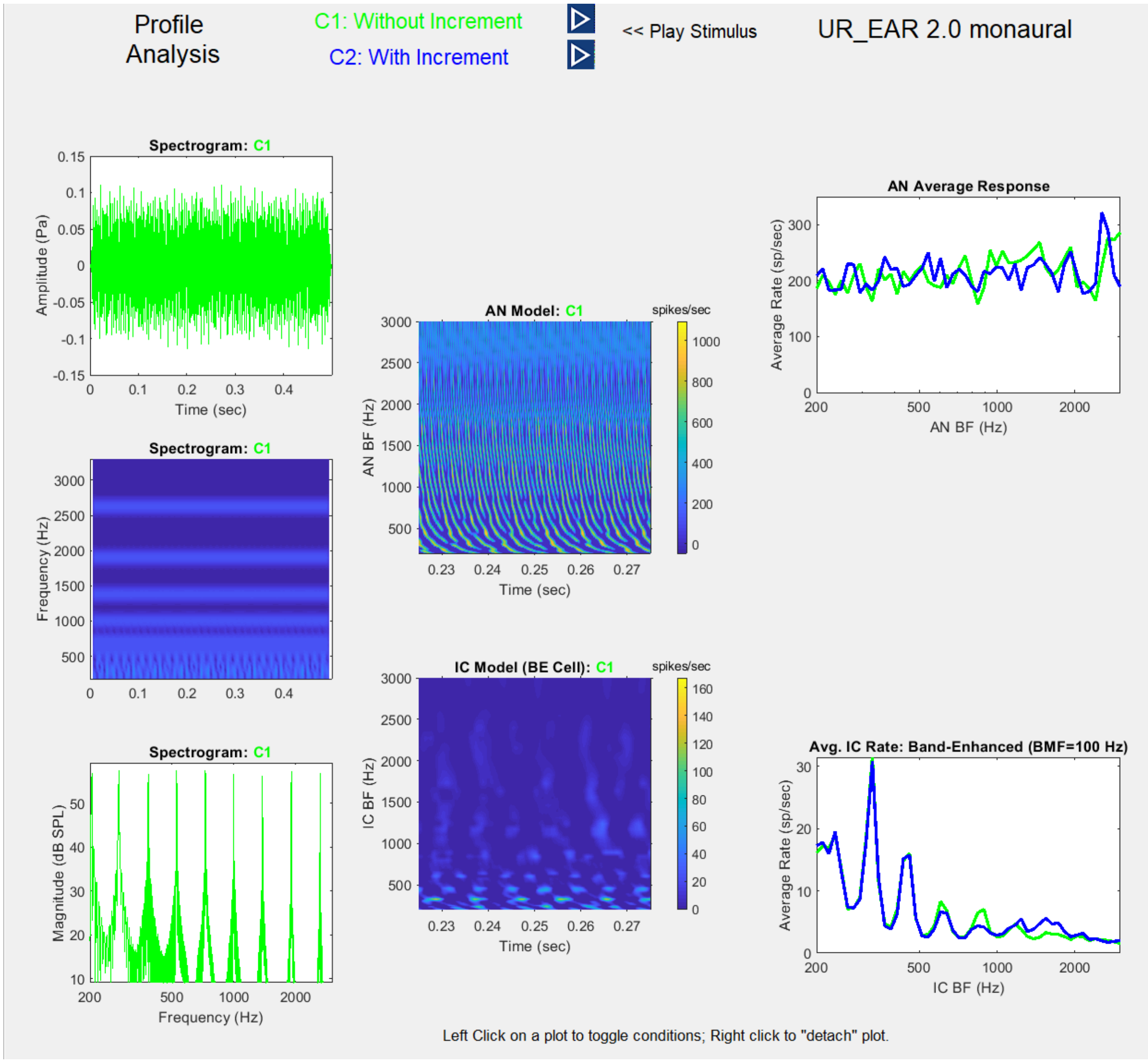
In Comodulation Masking Release (CMR) the Gaussian masker noise is modulated with an envelope created by a low-pass filtered noise (shifted up by a constant). Modulation results in lower detection threshold as bandwidth (BW) of the masker noise increases. Two stimuli are unmodulated and modulated Gaussian masker noise with tone. You can change tone properties and the bandwidth of the masker in stimulus design. Similar to stimuli in Haggard, Hall & Grose 1990.

CMR-FB:

These stimuli consist of a tone with narrow noise band centered on the tone frequency added to a series of other narrow noise bands. In one of the stimuli the flanking noise bands are modulated with the same frequency as the central noise band, which is shown in plots as the comod (comodulated bands) and in the other stimulus they are modulated with independent noise bands, shown by codev (codeviant bands). Similar to stimuli in Hall & Grose 1994.

Model Plots (Left panel of UR_EAR.m)

Interacting with the plots is fairly simple. This plot generator showcases comparisons between stimuli with slight variations, such as noise-plus-tone compared to noise alone. Here is an example model plot response:

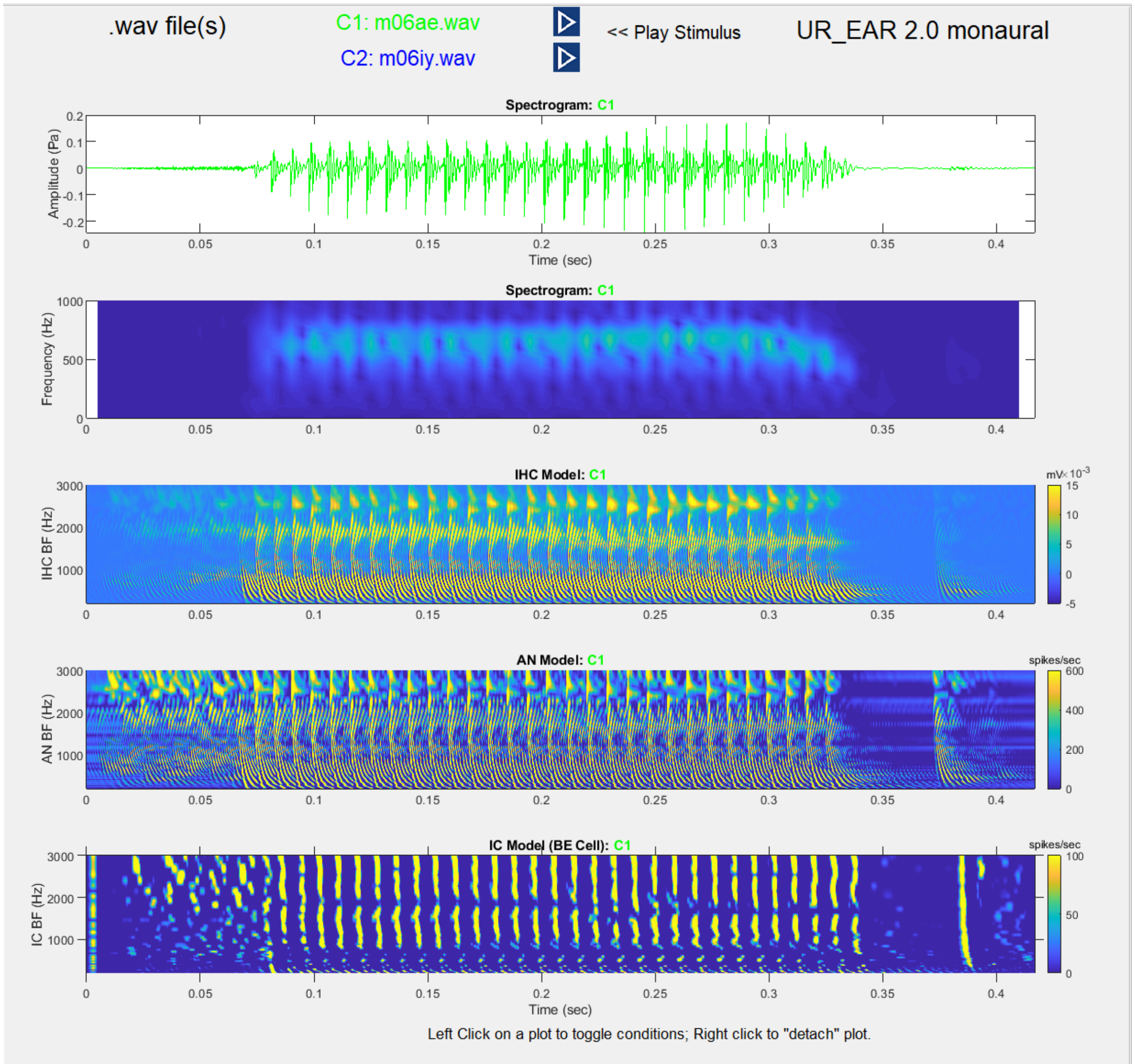


The left-most column of plots in the panel is the stimulus details, to allow the user to check if the stimulus is what they expected. The top left plot is the stimulus waveform for the selected stimulus. The second plot in the leftmost column is the spectrogram of the stimulus. The bottom plot in the leftmost column is the spectrum centered over the selected characteristic frequency (CF) range. The top middle plot is the auditory-nerve population model response over time. The bottom middle plot is the inferior colliculus population model response over time. The top rightmost plot is the auditory-nerve average spike rate as a function of best frequency. The bottom rightmost plot is the inferior colliculus band-enhanced model average population discharge rate as a function of best frequency. If the SFIE model is run, the middle rightmost plot is the cochlear nucleus model response.

The legend at the top shows the various conditions run (1 or 2), with a “play” button located next to each condition's name. If you left-click on a plot, it will toggle between conditions. If you right-click on a plot, you can detach that one plot from the entire response figure, a useful tool for saving and editing individual plots.

Each model you add will have to output plots into this panel. If you add a model yourself, you can decide which plots you’d like to utilize, change functionality, and more.

In wide display mode the first two plots are stimuli waveform and spectrogram respectively. The third plot is the Inner hair cell population model response over time. The last two plots are auditory-nerve and inferior colliculus population model responses over time respectively. You can see an example of wide display responses in figure 3.



Chapter 3 - Adding a New Stimulus

Introduction

To add a new stimulus, you need to follow these three steps:

1. ***Create your own function*** for producing your stimulus.

2. ***Edit UR_EAR***_v#_#.m to add your stimulus.

You will be working with UR_EAR_v#_#.m. The code itself is fully commented, so read through the m-file for a better understanding of the flow of the code and how to add to it. The code is a programmatic GUI designed to be edited. There are multiple panels within the GUI, such as a parameter selection panel, and a model plots panel.

1. Creating your own stimulus function

You need to create a function in its own .m file to add your stimulus to UR_EAR. Put your file in the same folder in order for UR_EAR to call your function. The inputs for your function need to be named specifically according to the UR_EAR code. Start with the most similar existing example! Your function must create the stimulus, convert the output to pascals, and ramp the stimulus on and off (in order to remove any artifacts resulting from abrupt starts or stops.

2. Edit UR_EAR Stimulus Section

Going through UR_EAR_v#_#.m, search for lines with the term "STIM" in them to spot places that require editing.

Chapter 4 - Adding a New IC or AN Model

Introduction

To add a new model, you need to follow these three steps (similar to adding a new stimulus):
You will be working with UR_EAR_v#_#.m. The code itself is fully commented, so read through the m-file for a better understanding of the flow of the code and how to add to it. The code is a programmatic GUI designed to be edited. There are multiple panels within the GUI, such as a parameter selection panel, and a model plots panel.

1. Adding Your Model Code

To add a model, you will need to add the code to the folder containing the UR_EAR folders and files. You will want to keep your model as a separate m-file. See the next two steps and the code itself for more details.

2. Adding to UR_EAR.m

Going through UR_EAR, you will need to do the following:

1) If your model is an IC model you need to add it to `modelTypeOptions` list and for AN models you should add it to `ANmodelTypeOptions list.`

2) `modelParamPanelUpdate`

You need to add a new `uipanel` here for your model. Keep most parameters the same as the existing models, except change the name. This will set up the panel where your model parameters will be selectable.

3) `Set up ___modelParamPanel components`

You need to set up a section for your model that creates the parameter options inside your model's `ParamPanel (uipanel)`. Use `SFIEmodel` and `ANICmodel` as examples, and note:

Keep `modelTypeDropDown {#} =` . Just change the number inside the { }, and change the name of the panel to the name you used for your model panel.

4) `Function modelTypePopUpCallback (~, ~, thisPanel)`

Here you need to add a new .visible for your model, then add a new `showThis` case. This will create a dropdown option for your model in the model panel of the GUI.

5) `Model Selection and Parameters`

This section is where the various different models are called and parameters for those models are determined. First there is a switch case to determine which AN model to use, and which IC model to use. If your model does not work with these, you'll need to set up a new case for the switch statement, or find another way to switch to your model.

8) `For iCF = 1:length(CFs);`

Here is the main set of loops to switch through models, allow different conditions to be plotted at once, and essentially allow the stimuli and models to interact with each other. You will want to go through this section of the code and its comments thoroughly, and try adding in your own model into the framework.

9) Overall

UR_EAR plotting panel calls stimuli and sets them up as structures, and selects models to run the stimuli through, and saves the output as a structure that is then plotted in various ways.

In general, you may need to add callbacks/callback info, change other parameters, and play around with getting your model to successfully show up in UR_EAR.

Chapter 5 - Quick plot of Single CF Responses

UR_Ear saves the AN and IC responses in the working directory, in a file called UR_EAR_model_data.mat. To look at the time responses (time-varying firing rate) for a single CF, choose the desire CF in the quick plot panel at the bottom of GUI and click on the quick plot button. The Quick Plot shows responses to the fiber in the population that is closest to the requested CF. Very simple, but sometime helpful for examining and/or illustrating the detailed responses.

References

Bruce, I. C., Erfani, Y., & Zilany, M. S. (2018). A phenomenological model of the synapse between the inner hair cell and auditory nerve: Implications of limited neurotransmitter release sites. Hearing research, 360, 40-54.

Krips, R., & Furst, M. (2009). Stochastic properties of coincidence-detector neural cells. Neural computation, 21(9), 2524-2553.

Carney, L. H., Li, T., & McDonough, J. M. (2015). Speech Coding in the Brain: Representation of Vowel Formants by Midbrain Neurons Tuned to Sound Fluctuations. ENeuro, 2(4). doi:10.1523/eneuro.0004-15.2015

Mao, J., Vosoughi, A., & Carney, L. H. (2013). Predictions of diotic tone-in-noise detection based on a nonlinear optimal combination of energy, envelope, and fine-structure cues. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 134(1), 396. doi:10.1121/1.4807815

Nelson, P. C., & Carney, L. H. (2004). A phenomenological model of peripheral and central neural responses to amplitude-modulated tones. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 116(4), 2173. doi:10.1121/1.1784442

Zilany, M. S., Bruce, I. C., & Carney, L. H. (2014). Updated parameters and expanded simulation options for a model of the auditory periphery. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 135(1), 283-286. doi:10.1121/1.4837815

Green, D. M. (1983). Further studies of auditory profile analysis. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 73(4), 1260. doi:10.1121/1.389274

Klein, M. A. (1981). Binaural edge pitch. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 70(1), 51. doi:10.1121/1.386581

Lyzenga, J. (1997). Frequency discrimination of stylized synthetic vowels with a single formant. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 102(3), 1755. doi:10.1121/1.420085

Lyzenga, J. (1998). Frequency discrimination of stylized synthetic vowels with two formants. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 104(5), 2956. doi:10.1121/1.423878

Patterson, R. D. (1976). Auditory filter shapes derived with noise stimuli. The Journal of the Acoustical Society of America J. Acoust. Soc. Am., 59(3), 640. doi:10.1121/1.380914

Schwarz, Doug. make_detachable function.

Contact Us

For more information, please visit our lab website at www.urmc.rochester.edu/labs/Carney-Lab/. You will find contact information, links to other code packages and literature, and more. E-mail inquiries should be directed to Laurel.Carney@Rochester.edu