UR EAR Version 2022a - User Manual

A guide to our software tool: “University of Rochester: Envisioning Auditory Responses” (UR_EAR Version 2022a). For modeling, creating stimuli, and more.

This manual pertains to:

A webapp located at https://URhear.URMC.Rochester.edu

Source code available at https://osf.io/6bsnt

Stand-alone versions for Windows, macOS, and GNU/Linux also available at https://osf.io/6bsnt

Note: The above source code and stand-alone versions require MATLAB2020B or later. Legacy versions (UR_EAR_2020b) that only require MATLAB2015 or later are available at the OSF website.

Table of Contents

1. Exploring the UR_EAR Package

Files and Contents of the Package

This MATLAB code package is designed to display auditory-nerve and midbrain response models given a collection of different stimuli. The UR_EAR Package contains one MATLAB app file (.mlapp), several MATLAB functions (.m files), and some C code for the innerworkings of the Auditory Nerve Model. Finally, MATLAB mex files are included, which are “MATLAB executables” containing compiled C code.

A list of the key files follows:

The main m-file for the GUI (Graphical User Interface) code:

    UR_EAR_2022a.mlapp

A folder containing Stimulus functions (see “Stimulus Functions” section below)

MATLAB “mex” files for the Zilany et al. (2014) model:

    model_IHC (mex file versions for Windows, macOS, and GNU/Linux)
    model_Synapse (Win/Mac/Linux mex files)

MATLAB “mex” files for the Bruce et al. (2018) model:

    Model_IHC_BEZ2018 (Win/Mac/Linux mex files)
    model_Synapse_BEZ2018 (Win/Mac/Linux mex files)

MATLAB functions that are used by both AN models:

    ffGn_ur_ear.m
    fitaudiogram2.m
    generate_neurogram_UREAR2.m
    generateANpopulation.m

MATLAB functions associated with the IC Models:

    SFIE_BE_BS_BMF.m
    get_alpha_norm.m
    unitgain_bpFilter.m

And Documentation:

    manual_UR_EAR_2022a.pdf
    readme_2022a.txt

Stimulus Functions

Each stimulus type is created by a function that is called by the UR_EAR app. The parameters are described within the function m-files. The most general form of a stimulus, with only the overall sound level as a variable, is an audio file (e.g., .wav files or other audio file types). For more information on how to add your own stimulus along with controllable parameters, see Section 3, "Adding New Stimuli".

In order to prevent interference with identically-named functions already in use, the stimulus functions are placed in a MATLAB "package folder" called +stimuli. **Do not put this folder on the MATLAB path.** If you want to use any of these functions in your own code outside of UR_EAR, you must follow the steps below for using a package:

1. First, import the stimulus package (without the +, as shown below)
2. Use the function as a typical .m file

```
import stimuli.*
waveform = TIN(…); %Note: TIN is an example stimulus function
```

Another method is to precede the function name with the package name (not including the + symbol) and a period. For example, to run the stimulus function, TIN.m, your code would be:

```
waveform = stimuli.TIN(...);
```

Note that this method is used in one place in UR_EAR, before the package has been imported. Consult the MATLAB documentation for "`import`" for more details on this procedure.

The stimulus m-files themselves are ordinary MATLAB functions; they are simply accessed a bit differently until the stimulus package has been imported. If you happen to have a package folder called "+stimuli" already, you should change the name of the package folder in UR_EAR by renaming the folder and then changing the UR_EAR code to match. The package name ("stimuli") appears twice in the UR_EAR code: once as "`stimuli.TIN`" and once as "`import stimuli.*`".

2. Running and Using the GUI

The Basics

For the webapp, select the "UR EAR 2022a" app at https://URhear.URMC.Rochester.edu. For the source code version, download the UR_EAR_2022a zip file from https://osf.io/6bsnt and extract the contents. Open MATLAB and either add the UR_EAR_2022a folder to your MATLAB path (preferably at the end of the path) or change your current folder to the UR_EAR_2022a folder. Then run UR_EAR_2022a by typing "`UR_EAR_2022a`" at the command line. A GUI should pop up. To test the code with the default parameters, simply press "Run". The GUI allows you to choose from a variety of models, stimuli, and parameter values (see below), then see the resulting plots that describe the auditory-nerve and midbrain responses.

Stimulus Selection and Stimulus Parameters

At the top left of the GUI is a pop-up menu where you can select which stimulus you'd like to use. Once a stimulus is selected, the "Parameters" section below it will change based on your selection.

The Parameters section is where you change the values of the parameters specific to the chosen stimulus. You can change the values of duration, frequencies of spectral edges or bandwidths, choose 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 "Run" to run the model. To compare various parameter changes that are not included automatically, simply run the model twice with those different selections.

Model Parameters

In the upper, middle section of the GUI is the "Model Parameters" section, where you select which auditory-nerve (AN) model and inferior colliculus (IC) model you'd like to use from the drop-down 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 center-frequency range, number of characteristic-frequency (CF) channels (evenly log-spaced across the designated frequency range), the number of AN fibers at each CF, and hearing status (by audiogram).

To simulate responses of an ear with sensorineural hearing loss, input the hearing loss in dB HL at each of the audiometric frequencies, as described in Zilany and Bruce (2007). For normal hearing, all of the values should be zero.

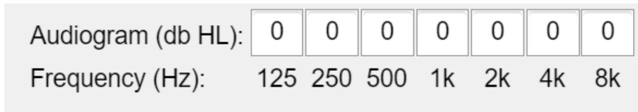


Figure 1: Hearing Loss Parameter Input Boxes

In UR_EAR_2022a, there are two different AN models and two different IC models from which to choose: Zilany et al. (2014) and Bruce et al. (2018). For each model, you will enter the range of CFs and number of channels. You will also select the species, cat or human, to determine the sharpness of tuning and the middle ear filter model. Cat parameters are from Zilany et al. (2014) and human parameters are from Ibrahim & Bruce (2010). Finally, you can select low, medium, or high spontaneous rate AN fibers on the GUI.

The two IC models are the Same-Frequency Inhibition and Excitation Model (SFIE, Nelson & Carney 2004), and the simpler Inferior Colliculus (IC) Modulation Filter Model (Mao et al. 2013). For both models, you need to specify the best modulation frequency (BMF). For the SFIE model, the responses are shown for either band-enhanced (BE) or band-suppressed (BS) model neurons, selectable by a radio button.

Output Plots

There are two sets of plots. In the middle section of the GUI there are plots of the stimuli (left column), neurograms (middle column), and average rate profiles (right column). The stimulus plots include a waveform, a spectrogram, and a spectrum over the range of CFs that were specified. For simulations of two stimuli, C1 and C2, you can toggle between the two stimuli by clicking on any of the stimulus or neurogram plots. The neurograms show a 2D plot of time-varying discharge rates for the population of AN fibers (top) and IC neurons (bottom) over the analysis window.

The AN model and IC model plots are shown for only a portion of the full duration. This time span is called the analysis window and can be adjusted by left-clicking and dragging one or both of the vertical gray lines appearing in the stimulus waveform plot. Right-click and drag to move both lines together, keeping the same span between them. You can also enter explicit time values into a dialog box by clicking on the analysis window display text at the upper right corner of the waveform plot. By default, the analysis window will be reset to the default values each time you ‘Run’ the model. Below this are three buttons. “Reset” will reset the analysis window to the default (10% and 90% points of the shorter stimulus), “Help…” brings up a dialog box with detailed analysis window instructions, and “Hold” (in red) will toggle the Hold state. When “on,” the current analysis window will not be reset when running the model. Helpful hint: if you accidentally move a gray line to a place where it is no longer visible, you can recover it by clicking on “Reset.”

The averages rates shown in the third column of model output plots are computed only for the time span of the analysis window, and you can see these plots change as you adjust the window. The average discharge rate plots (right) for AN and IC show the average rate for each CF, average across the set of fibers in each CF channel, over the analysis window. When there are two lines in a plot, one is for each stimulus, color coded to match the stimulus waveform. The bottom section of the GUI shows the wide display of all of the outputs over the entire stimulus waveform.

Right-clicking on a plot will bring up a menu containing a number of options for saving the plot or the data that produced it.

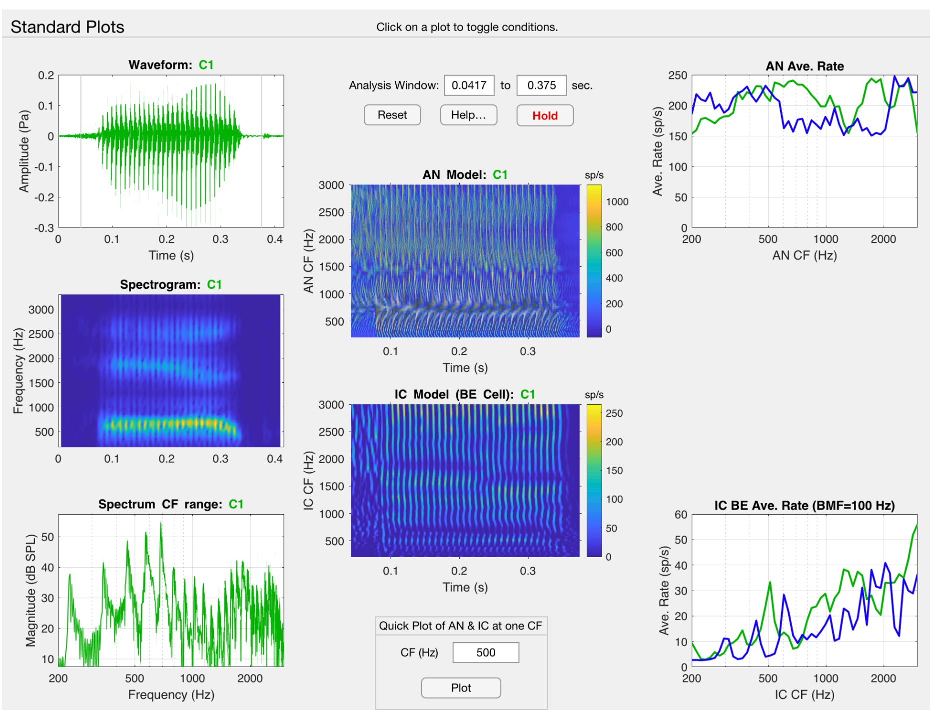


Figure 2: Standard Model Plots with Default Parameters

Quick Plot

In the bottom middle region of the Standard Plots panel is “Quick Plot”. This function will open a dialog box containing single-CF AN-model and IC-model plots from the responses shown immediately above, shown as typical 2-D plots of the instantaneous rate functions. Enter the desired CF and press the Plot button; the function will choose the nearest CF in the model population and produce the appropriate plots.

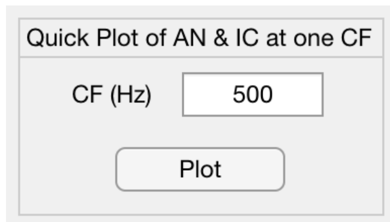
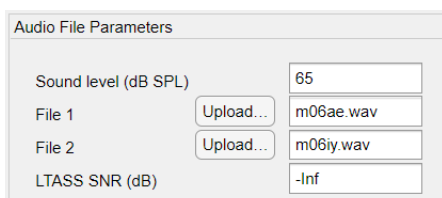


Figure 3: Quick Plot panel.

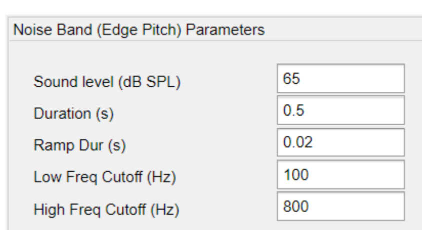
The Built-In Stimuli

This software includes many built-in stimuli ready to interact with the built-in models. The stimuli are based on those used in psychophysical or physiological studies. All stimulus waveforms are in pascals, which is important because of the nonlinearity of the AN model. The sound level is the overall level, and the stimulus duration includes $\cos^2$ onset/offset ramps at the specified ramp duration. Following is a list of default stimuli and their descriptions:

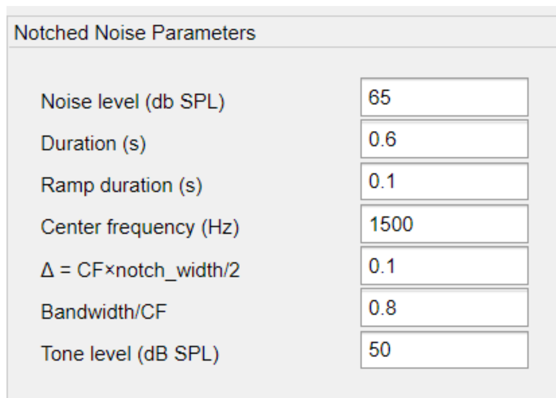
- **Audio file:** Almost any type of audio file can be used as a stimulus. We have included two example files as the default: vowel-consonant-vowels, 'm06ae.wav' ("had") and 'm06iy.wav' ("heed") (from the Hillenbrand et al. 1995 database). You can upload an audio file from your home computer as the stimulus using the Upload button. BEWARE: if you read in a very long sound file, the simulations will take a long time! You must set the overall sound level in dB SPL to scale the signal into pascals. If noise is desired, you can set the SNR for LTASS noise (Long Term Average Speech Spectrum). If the sound level entry is left blank, the noise field becomes noise level in dB SPL instead of SNR, and the audio-file waveform is not scaled in any way, therefore, the user should provide this waveform already scaled into pascals.



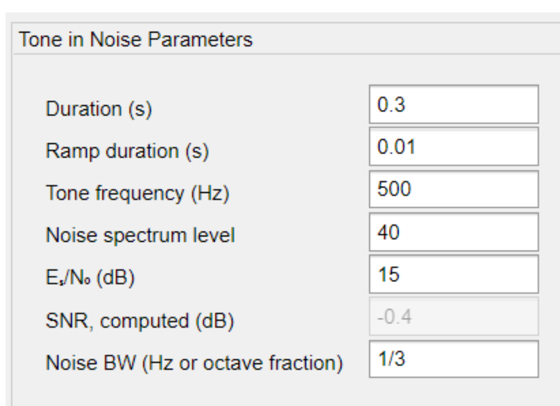
- **Noise Band (Edge Pitch):** Gaussian noise stimulus, with a band specified by selecting the low and high frequency limits. This stimulus is useful for exploring *Edge Pitch,* based on Klein & Hartmann (1981). A pitch is perceived at about 4% inside the noise band from a sharp spectral edge. The IC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



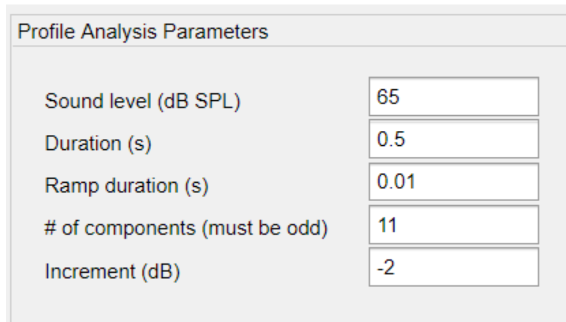
- **Notched Noise:** Notched noise is a commonly used masking stimulus for psychophysical experiments, such as in Patterson (1976). The stimulus consists of two bands of noise with a notch centered at the signal (center) frequency. The notch width is specified by Δ. The width of each noise band is specified in terms of bandwidth/CF. Two stimuli will be created: one is the notched noise alone and the other is notched noise plus a tone at the center frequency and at the tone level. See Maxwell et al., (2020) for a study of this stimulus.



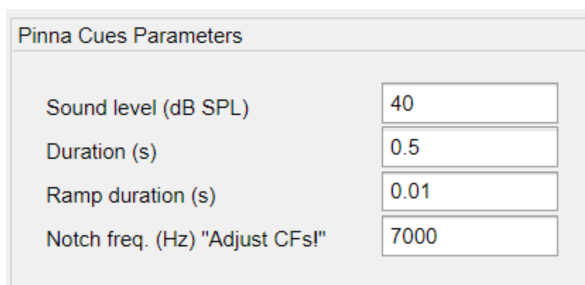
- **Tone in Noise:** Another popular research topic in auditory processing, tone-in-noise paradigms involve discerning the difference between a stimulus of just noise, and of the same noise with an added tone. The noise bandwidth can be specified by typing two frequencies, or a fraction of an octave by typing a fraction with a slash (/). For example, “1/3” is interpreted as ⅓-octave noise centered on the specified tone frequency. If you want 1-octave noise, type “1/1”. The noise spectrum level is in dB SPL/Hz. The SNR is specified by the E_s/N_0 and the SNR (the ratio of overall tone level to overall noise level) is also computed and displayed.



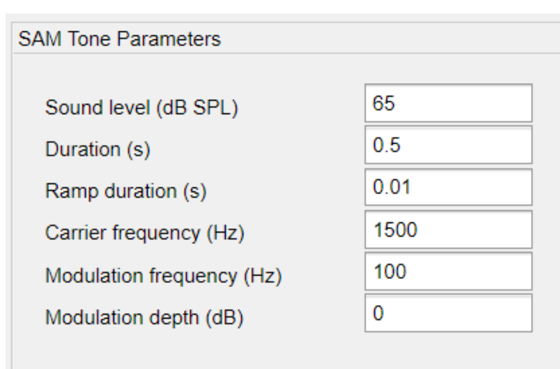
- **Profile Analysis:** This stimulus is a waveform constructed of evenly log-spaced components centered around 1,000 Hz. This stimulus can be presented with and without an increment of the center frequency component. The number of components can be specified. See Green (1983) for details.



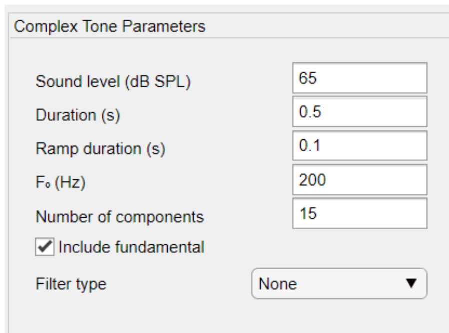
- **Pinna Cues:** Filters band-limited noises using an artificial notch, modeling head-related transfer functions that simulates a simple spectral cue for sound localization. The notch frequency is selected; be sure to adjust your CF range to span the notch frequency.



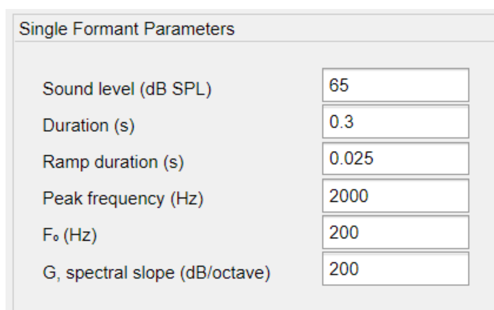
- **SAM Tone (and pure tone):** Creates both a pure tone and an AM tone with specified carrier and modulation frequency and depth. Modulation depth is $20*\log_{10}(m)$, where m is the modulation index.



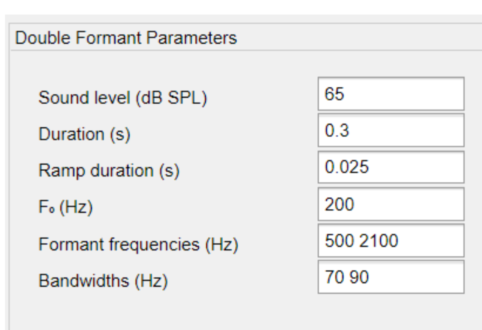
- **Complex Tone / Harmonic Complex:** This stimulus is a complex of equal amplitude tone components spaced by F_0 Hz. Specify the total number of components and whether or not to include the fundamental frequency. A filter can be selected to limit the range of components to low frequencies, high frequencies, or a band. After you select a filter, you will be asked to input the appropriate cutoff frequencies.



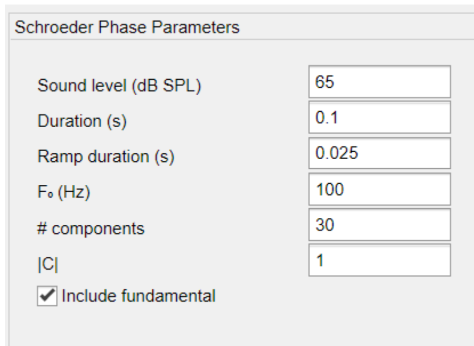
- **Single Formant:** The stimulus is a harmonic complex with a spectral envelope that is a simple triangular spectral peak, similar to those seen in vowels, à la Lyzenga & Horst (1995). Specify the peak frequency and the F_0, and G is the slope of the spectral envelope.



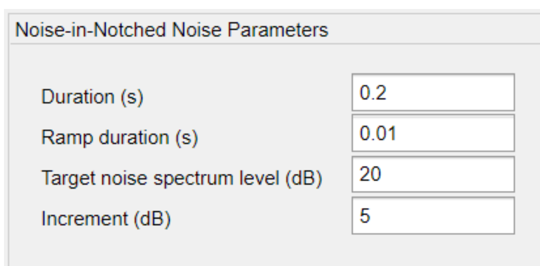
- **Double Formant:** This stimulus is similar to the single format, except that the spectral envelope has two formant peaks, generated using Klatt resonators. Similar to stimuli in Lyzenga & Horst (1997,1998).



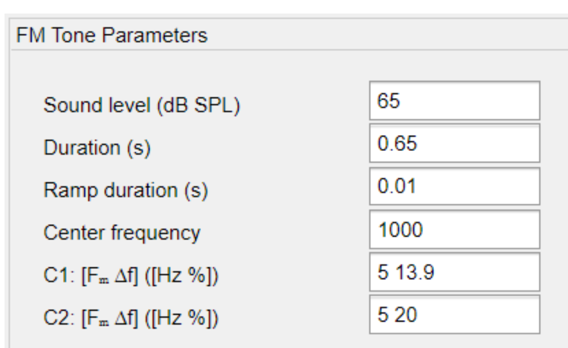
- **Schroeder Phase Harmonic Complex:** This stimulus is a harmonic complex with a phase spectrum that results in a flat envelope (for C = ±1). Set F_0 and number of components. Stimuli with positive and Negative C values are created. See Schroeder (1970) for details.



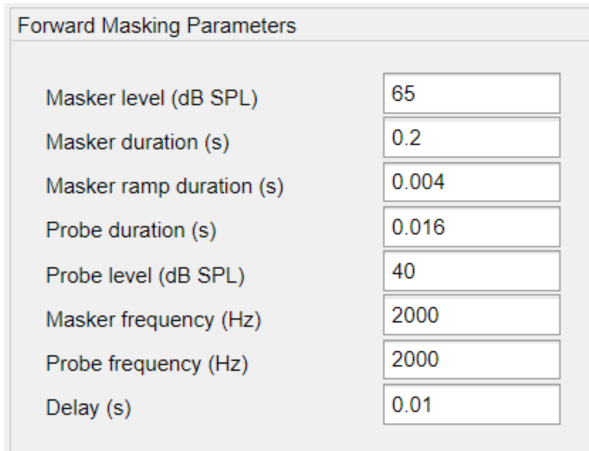
- **Noise-in-Notched Noise:** A band of noise (target noise) is added to a complementary band-reject noise (masker noise) which is 10 dB higher in spectrum level. In test stimulus the target noise spectrum level is increased and in the standard stimulus the target noise level is fixed. Discrimination between these two stimuli was studied in Viemeister (1983) and Richards & Carney (2019). For stimulus details see these papers.



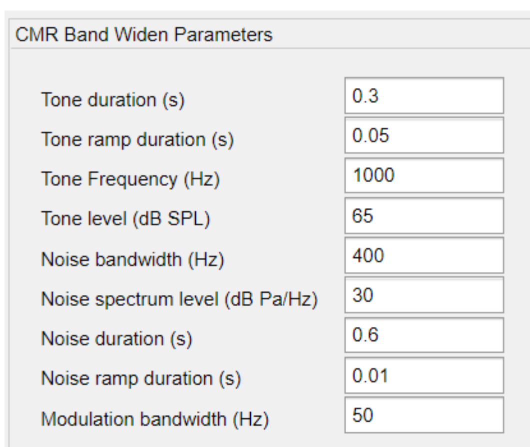
- **FM tone:** Two sinusoidal frequency modulated tones (C1, C2) with two different modulation frequencies (Hz) and modulation excursions (frequency sweep ranges, specified as % of the center frequency).



- **Forward Masking:** First condition is a pure-tone masker with a short delay before the probe tone, and second condition is just the pure-tone masker. You can change properties of both masker and tone in the stimulus parameters in GUI.



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



- **Co-modulation Masking Release (CMR) – Flanking-Band Stimuli:** These stimuli consist of a tone plus a narrow-noise band centered on the tone frequency, added to a set of other 'flanking' narrow-bands of noise. In one of the stimuli the flanking noise bands are modulated by the same noisy envelope as the central noise band (i.e., 'co-mod' or comodulated bands). In the other stimulus, the flanking bands are modulated with independent noise envelopes (i.e., 'codev', or codeviant bands). Similar to stimuli in Hall & Grose (1994).

CMR Flank Band Parameters

Duration (s)	0.3
Ramp duration (s)	0.01
Tone Frequency (Hz)	1000
Noise spectrum level (dB Pa/Hz)	40
Tone level (dB SPL)	65
Flank bandwidth (Hz)	100

3. Adding New Stimuli

To add a new stimulus, you need to **Create Your Own Function** and **Edit UR_EAR_2022a.mlapp.** You can only do this if you are using the .mlapp version, i.e., not the web app or standalone executable.

Create Your Own Function

You need to create a function in its own m-file to add your stimulus to UR_EAR. You can put your file in the +stimuli subfolder, but this is not necessary as long as MATLAB can find it, e.g., on the MATLAB path. Your function must get its inputs from parameters typed into the GUI. You can see the built-in stimulus functions as examples. Start with the most similar existing one! 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). The MATLAB tukeywin function is a convenient way to generate a raised-cosine ramp.

Edit UR_EAR_2022a.mlapp

Going through UR_EAR_2022a.mlapp, search for lines with the term “STIM” in them to spot places that require editing to accommodate your new stimulus. The code itself is fully commented, so read through it for a better understanding of the flow of the code and how to add to it. The code was created using App Designer and intended to be edited. There are multiple panels within the GUI, such as a parameter selection panel, and a model plots panel.

4. Adding a New Auditory-Nerve Fiber or Inferior Colliculus Model

Introduction

Adding a new model consists of these two steps: modifying the app so that your model can be selected, and modifying model.m to compute your model. The code itself is fully commented, so read through it for a better understanding of the flow of the code and how to add to it. The app code was created using App Designer and model.m is a regular m-file. Both are intended to be edited. There are multiple panels within the GUI, such as a parameter selection panel, and a model plots panel.

Changes to the app

Inside the app method "`createModelParamPanel`", modify the variable "`ANmodelTypeOptions`" (for a new AN model) to include the name of your model or modify the variable "`ICmodelTypeOptions`" (for an IC 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.

Changes to model.m

For an AN model, add a case to the "switch Which_AN" construct. For an IC model, add a case to the "switch Which_IC" construct. Use the built-in model code as a guide. Your model code must work on one CF at a time and you must not change the model.m arguments.

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.

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.

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.

Haggard, M. P., Hall III, J. W., & Grose, J. H. (1990). Comodulation masking release as a function of bandwidth and test frequency. The Journal of the Acoustical Society of America, 88(1), 113-118.

Hall III, J. W., & Grose, J. H. (1994). Development of temporal resolution in children as measured by the temporal modulation transfer function. The Journal of the Acoustical Society of America, 96(1), 150-154.

Ibrahim, R. A., & Bruce, I. C. (2010). Effects of peripheral tuning on the auditory nerve's representation of speech envelope and temporal fine structure cues. In The neurophysiological bases of auditory perception (pp. 429-438). Springer, New York, NY.

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.

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

Lyzenga, J., & Horst, J. W. (1995). Frequency discrimination of bandlimited harmonic complexes related to vowel formants. The Journal of the Acoustical Society of America, 98(4), 1943-1955.

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.

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.

Maxwell, B. N., Richards, V. M., & Carney, L. H. (2020). Neural fluctuation cues for simultaneous notched-noise masking and profile-analysis tasks: Insights from model midbrain responses. The Journal of the Acoustical Society of America, 147(5), 3523-3537.

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.

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.

Richards, V. M., & Carney, L. H. (2019). Potential cues for the “level discrimination” of a noise band in the presence of flanking bands. The Journal of the Acoustical Society of America, 145(5), EL442-EL448.

Schroeder, M. (1970). Synthesis of low-peak-factor signals and binary sequences with low autocorrelation (Corresp.). IEEE Transactions on Information Theory, 16(1), 85-89.

Viemeister, N. F. (1983). Auditory intensity discrimination at high frequencies in the presence of noise. Science, 221(4616), 1206-1208.

Zilany, M. S., & Bruce, I. C. (2007). Representation of the vowel/ε/in normal and impaired auditory nerve fibers: model predictions of responses in cats. The Journal of the Acoustical Society of America, 122(1), 402-417.

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.

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. While using the model, feedback can be sent directly to Laurel Carney via email using the “Contact…” button in the top right corner of the web app.