Auditory Filter Shapes

Introduction

In this lab you will have the opportunity to carry out psychoacoustic measurements on yourselves and to analyze the results. The experiment involves estimating the shape of the auditory filter at 1 kHz using an abbreviated version of the notched-noise technique, proposed by Patterson, and elaborated by Moore and others. The measurements will be done at two different noise levels, to test for any changes in filter shape with level.

The experiments should be carried out in pairs, so that the setting up can be done together. You can then take it in turns to run the experiment. Breaks every 15 minutes or so are recommended

The analysis may also be carried out in pairs, so if you have no experience with Matlab, or are a little shaky on calculus, team up with someone who knows about these things!

The writing-up should be done alone. If you needed help with the analysis, please state this in the lab report. It will not be held against you. However, the correct person should be acknowledged for any ingenious solutions to the problem(s).

Running the Experiment

Only five data points will be used to estimate the filter shape at each of two noise levels, but each point should be repeated once, giving a total of 20 runs (10 at each noise level). One session of about 2-3 hours per pair should be sufficient to carry out the experiment

The Set-up

The waveforms are created on a PC using Matlab. Sound is generated from those waveforms using a 24-bit digital-to-analog converter (DAC) in the PC. The electrical signal is then fed via a headphone buffer (TDT HB6) to the booth. In the booth, the stimuli are presented via headphones.

Before you start the experiment, it is very important to make sure that the wiring and the attenuation settings are correct. Make sure the HB6 switch is set to -10 dB (‘up’ position).

Start up Matlab.
There should be a handheld voltage meter outside the booth. Use this to verify

the voltage at the headphone amplifier left output. To check the voltage enter:

> calibrate('mid',-10,'l')

This tells the system that you have 10 dB attenuation in the path (from the headphone amplifier) and that you are testing the left channel only (' l').

On the screen will then appear the voltage you should expect to measure at the output to the headphone buffer. Check that the actual voltage does not differ from the predicted voltage by more than about 10%. (Remember that a barely detectable 1-dB change is already 12%).

The Experiment

Stimuli

The object is to detect a 1-kHz signal in the presence of a white noise with a bandwidth extending from 20 to 10000 Hz and a spectrum level of either 15 or 40 dB SPL. The experimental parameter is the width of a spectral notch, centered around the signal frequency. The notch widths on each side of the signal frequency are (a) 0 and 0 (no notch); (b) 0.1 and 0.1; (c) 0.2 and 0.2; (d) 0.3 and 0.3; and (e) 0.4 and 0.4. These widths are specified relative to the signal frequency, e.g., 0.2 and 0.2 refers to a width of 200 Hz on each side, producing a notch from 800 to 1200 Hz.

Method

Each trial consists of three 200-ms noise bursts. The task is to decide which of the three intervals also contains a 150-ms, 1-kHz signal, which is temporally centered in the noise. All the stimuli are gated with 10-ms ramps to avoid spectral "splatter". This is a 3-interval, 3-alternative forced-choice procedure. The signal level begins at what should be an easily detectable level and is varied adaptively according to a 2-down 1-up rule, tracking the 70.7% correct point on the psychometric function. The signal level is initially varied in steps of 8 dB. After the first two reversals, the step size is reduced to 4 dB. After a further two reversals, the step size is reduced to its minimum value of 2 dB. Each run is terminated after six more reversals, and the threshold is defined as the mean level at the last six reversals. Two repetitions of each condition will be run. The presentation order of the conditions is randomized by the program.

Data storage

The data are stored in a file named notch_ *xx* _1k.dat, where *xx* should be your initials. When you start this experiment, a new file called control_notch_ *xx* _1k.dat is created. This is a list of all the conditions that will be tested and is used by the program to determine what condition to test next.

If you decide to do a few practice runs first (a good idea), you MUST delete your control and data files before starting the 'real' experiment.

Starting the experiment

To start this experiment, enter the following line in the Matlab command window:

>   afc_main('notch','xx','mid','0','1k')

Note that all the arguments are in single quotes and are separated by commas. The first argument is the experiment name; the second is for your initials (e.g., 'js' for John Smith); the third is the booth name ('mid' in this case), the fourth is the amount of attenuation set on the TDT PA4 (0 because the PA4s are not included in the circuit), and the fifth is the signal frequency.

This should result in a virtual response box appearing, which gives you instructions about what to do next. At the end of each run you have the option to start a new run, or to end the session. If you end or if you interrupt the program (by, for instance, closing the response window), you can start where you left off simply by reentering the line given above – the program will know what conditions you still have to do by looking at the control file.

DATA ANALYSIS

When you have completed all 20 runs, the program will stop automatically. However, you can check your progress at any time by opening your data file (notch_xx_1k.dat). In the data file, the first column is the condition. This is given as a number, where the first digit is the notch on the lower side, and the second digit is the notch of the upper side of the signal frequency. For instance, condition 44(.0000) means a notch of 0.4 below and 0.4 above. The second column is the level of the signal at threshold (in dB SPL). The third column is the standard deviation of the last six reversal points. Copy your data file ('notch_xx_1k.dat') on to a diskette or ftp it to a location of your choice. It is in ASCII text so it should be readable in whatever application you choose to do the analysis.

Stage 1: Individual data analysis

Plot the individual data points on a graph, with signal threshold vs. notch width, to give you a feel for how thresholds change with increasing notch width. Are there noticeable differences in the raw data between the two masker levels?

Next, think about finding a filter function suitable for describing your data, within the context of the power spectrum model. To make life easier for you, only symmetric notches have been used, meaning it has to be assumed that the filter is symmetric. In effect, therefore, you may ignore one half of the filter function.

Remember that the signal threshold is supposed to be proportional to the noise power passed through the filter.

There are two aspects to the filter. The first is the filter shape. This can be any arbitrary function, with sufficiently many (but not too many) free parameters to fit the data. The shape of the filter will determine how predictions change as the width of the notch changes. The second aspect is filter efficiency (K). This is the signal-to-noise ratio at threshold and provides a vertical shift of the predictions.

One popular filter function is known as the roex (rounded exponential) function. This is described by:

$$W(g) = (1- r)(1+ p|g|)\exp(- p|g|) + r,$$

where g is the normalized frequency deviation [$(f–f_s)/f_s$, where f_s is the signal frequency], p determines the slope of the filter, and r limits the dynamic range of the filter. This is known as the roex(p,r) model. Remember that W(g) is in the power spectrum (not the amplitude spectrum) domain (i.e., use 10log, not 20log).

According to the power-spectrum model, your thresholds estimates are proportional to the noise energy falling within the filter, so that the signal-to-noise ratio at the output of the filter should be the same in all conditions. The Appendix of Patterson et al. (1982) explains how a filter shape can be derived.

One way of deriving a reasonable fit is to simply try out different combinations of parameters and look at the effects. A more systematic way is to implement a least-squares minimization routine, such as Matlab's 'fminsearch'. Whichever way you decide to approach the problem, explain your approach and assumptions thoroughly. Plot your data together with your filter predictions and (in a separate figure) the resulting filter shape. Calculate the equivalent rectangular bandwidth of your derived filter at both masker levels.

Stage 2: Mean data analysis

Once everyone has completed their measurements, I will e-mail you a file containing the data from everyone. Find a sensible way to combine them into a summary graph (e.g, calculate the mean and s.d. for each condition). Using the technique you used for your own data, fit a filter shape to the mean data at both masker levels and calculate the ERB for each. How does the mean data compare with values from the literature (Eq. 9, p. 175 in (Moore, 1995) *Hearing* , 1995, ed. B.C.J. Moore)? How well does the model describe your data?

WRITING UP

Your lab report should describe the experiment, the results, and the modeling, concentrating on the latter two aspects. Include plots of the data, the model predictions, and the derived filter shape itself. How well does the model describe your data? Finish with a brief discussion of the assumptions underlying the model and their validity.

References

Moore, B. C. J. (1995). Handbook of Perception and Cognition, Volume 6. Hearing

Patterson, R. D., Nimmo-Smith, I., Weber, D. L., and Milroy, R. (**1982**). The deterioration of hearing with age: frequency selectivity, the critical ratio, the audiogram, and speech threshold. *J. Acoust. Soc. Am.* **72**: 1788-1803.