Human Auditory Periphery (HAP)

Ray Meddis

Department of Human Sciences, University of Essex Colchester, CO4 3SQ, UK. rmeddis@essex.ac.uk

A demonstrator for a human auditory modelling approach.



1 Introduction

The human auditory periphery operates in a nonlinear fashion. Historically, its operation has been modelled using linear gammatone filters. New nonlinear models are now available. This program gives ready access to one of these models by producing a graphical display of the result of a nonlinear analysis of arbitrary .wav and .aif files. It also permits comparisons with traditional linear gammatone analysis. This particular package has been designed with the needs of the automatic speech recognition (ASR) community in mind.

If you experience any difficulties with this program please contact Ray Meddis, <u>rmeddis@essex.ac.uk</u>.

1.1 Contents

The programmes are contained in a folder humanAuditoryPeriphery. This folder contains

- Matlab demonstration programs to illustrate the use of the Human Auditory Periphery (HAP) software
- simulation and parameter files for use in conjunction with the underlying Auditory Modelling System (AMS) model
- Sample .wav files to demonstrate the use of the software

1.2 What it does

The input to the system is a short sound stimulus in a .wav (or .aif) file

The **output** from the system is the response of the nonlinear or linear model in the form of an excitation pattern varying in time. This is what you see displayed on the graphical user interface (GUI). This pattern is also saved as a text file, *output.dat*.

1.3 The model

The Human Auditory Periphery model consists of three simulation stages

- an outer/middle ear filter
- a nonlinear filter bank simulating the response of the basilar membrane
- a sliding temporal integrator

For more information, see the section below on The underlying model

1.4 How to use it

The model can be run in a number of different ways

1. As a dedicated user interface (MATLAB-GUI). See section <u>Using the HAP</u> interface

interface

 As a MATLAB function that converts a .wav file into an excitation pattern file (i.e., with *no* graphical user interface). See section <u>Calling HAP directly from</u> MATLAB. 3. As a stand-alone AMS graphical windows application that does not require MATLAB. See <u>section 4.</u>

1.5 Contributors

Many people have contributed to the development of the system and acknowledgements will accumulate as this documentation matures.

The GUI was written by Ray Meddis. Comments and suggestions should be sent to <u>rmeddis@essex.ac.uk</u>

The model was created using MATLAB and the Auditory Model Simulator (AMS) application. AMS was created by Lowel O'Mard specifically for modelling auditory function. More information can be found at

http://www.essex.ac.uk/psychology/hearinglab

Brian Moore, supplied the values for the outer/middle ear filter.

References to the authors of components of the model can be found in the section below on the <u>The underlying model</u>.

Using the HAP interface

1.6 Getting started.

You will need

- a PC with Windows 2000 operating system or later
- AMS . An installer for the latest version of AMS comes with this package. If AMS is already installed, you will not need to reinstall it.
- MATLAB version 6. This will run the interface or use the model as a callable function. The GUI was created using version 6.1.0.450 (R12.1).

1.7 Installation

The normal way to receive HAP is part of a self-installing package of auditory modelling material. If you have received the program in this way, HAP will be found in the folder $C:\DSAM\AMS\HAP$.

The user, however, needs to put the HAP program on the MATLAB fire path:

- 1. Launch MATLAB
 - \circ add the folder *C:\DSAM\AMS\\HAP* to the MATLAB path using the MATLAB pull-down menu (File/Set Path...).

1.8 Run HAP

Running instructions for the HAP interface: Open MATLAB

Type *HAP*.

This will launch the HAP interface panel that will allow you to interact with the model. For best effect, maximize the display to full-screen size using the maximize box to the right of the title bar.

Select a .wav file from the directory window (top right of the HAP GUI display).

A number of demonstration .wav files should be visible in the listbox. Use these while familiarising with the interface.

Double-click on the file name to initiate the HAP processing.

The selected sound will play.

If you can't hear it, the volume control of your PC may have been set to mute. If you don't want to hear this, set the volume control to 'mute'.

The excitation pattern will appear in the figure window when processing is complete. As an alternative to the 'double-click', first select the file and then click on 'run model'.

1.9 What are you looking at?

The x-axis of the figure is time and the y axis shows the centre frequencies (CFs) of an auditory filterbank.

The z-axis of the graph shows the output of the 'temporal window function'. This is moving average of the output of the nonlinear filters.

More information on the underlying model can be found below in the section on <u>The</u> <u>underlying model</u>.



1.10 Stop!

There is no simple way of stopping the analysis. This is not normally a problem as analysis times are typically short. However, if you have chosen a long file and are regretting it, click on the close box in the top right-hand corner of the display. This will close the GUI although MATLAB will remain active.

To restart, type HAP in the MATLAB command window.

Avoid using long files until you are familiar with the program.

1.11 More about the interface

1.11.1 Navigating using the listbox (directory)

The directory in the top right of the screen can be used to navigate to other folders where *.wav* files are stored.

- Double-click on the '..' symbol to move up to the enclosing folder
- Double-click on any folder name to open it

Only *.wav* and *.aif* files are shown. Any name without a *.wav* or *.aif* extension is a folder.



1.11.2 Pull-down menus

level (peak dB SPL)	70 💌	
min CF	100 💌	
max CF	3000 💌	
number of CFs	40 💌	
model type	nonLinear 🗾	
Run Model		

The pull-down menus below the directory box allow you to change the signal level, the number of channels and the range of centre frequencies (CFs). It also allows you to change between the new nonlinear model and a more traditional linear model.

Operation

- Select the required value from the pull-down menu
- click on the 'Run Model' push button to repeat the analysis with the new parameters.

Peak signal level.

- is in dB SPL
- HAP assumes that the .wav file is using its full dynamic range (-1 to +1). It rescales the signal to have a peak value (+1) with the number of microPascals appropriate for the peak value specified.
- The peak signal level is important for a nonlinear filterbank because the nonlinearity is level-dependent. Therefore, it needs to know the signal values in terms of microPascals. Unlike a linear system, the shape of the excitation pattern will vary with signal level.

Number of channels.

- To begin, only 20 channels are used.
- While experimenting with the controls, it is a good idea to keep the number of channels small. Increasing the number of channels will produce more interesting results but will take longer.
- Very large numbers of channels (and long sounds) may trigger the use of virtual memory and slow the operation considerably.

Linear/nonlinear choice

- The purpose of this demonstration is to introduce nonlinear models. The inclusion of a linear option is to permit comparison with previous models
- In general, nonlinear output is smoother than linear, particularly at higher signal levels.



Figure 1 Nonlinear (left) and linear (right) auditory representation of the word 'four' (male speaker) spoken with a peak signal level of 80 dB SPL. The nonlinear filters give a much flatter representation.

1.11.3 Display controls

The controls at the bottom left of the screen can be used to tailor the view as required. The image should update immediately after any change to these controls. Hopefully they are self-

Display control					
azimuth	-16	٩		Þ	🗹 dB scale
elevation	44	◀		Þ	🗹 ERB scale
select color map		su	summer 🗾 💌		🗹 flat shading

explanatory. Some points are worth noting however.

dB scale puts the excitation pattern on a vertical dB scale. Values below 0 dB are omitted. The dB scale has been roughly equated with the input dB SPL values for display values. The scale has no simple physical meaning but is related to 'perceived loudness'.

ERB scale arranges the display to give either a logarithmic representation of channel centre frequency, *CF*, or a linear representation. The logarithmic representation is, in fact, an equal spacing on the ERB scale. More on ERB scales can be found in Moore, BCJ (1989) an Introduction to the Psychology of Hearing, London: Academic Press. The example below is a female voice saying 'dah'.

N.B. the channel CFs themselves are *always* distributed on a log scale along the basilar membrane. Changing the display does not alter this fact. A linear display is a distortion of the model result but can be useful when looking for harmonic structure.



Figure 2 Comparison of ERB scale and linear scale display for speech file 'dah'.

Flat shading removes the mesh lines. This can be very useful if there are a large number of channels.

Azimuth and elevation. The viewpoint can be set using the azimuth and elevation controls.

- If the arrows at the ends of the sliders are continuously pressed, the display will rotate slowly.
- A colour contour plot can be obtained by setting the azimuth to zero and the elevation to 90.



Figure 3 Density plot obtained by setting azimuth to zero and elevation to 90.

1.11.4 Sound

Sound can be switched off or on using the sound check box at the bottom of the display.

1.11.5 Input files

- At present, HAP is set up to read only .wav and .aif files.
- Keep them short or expect to wait!
- If the files have low sample rates, HAP endeavours to up-rate the sampling rate to approx. 50000 Hz by over-sampling. This is to satisfy the requirements of the filtering processes. Using low sample rates will therefore not speed up the filtering process.

1.11.6 Listen to the selected file

You can listen to a file before or after processing it.

- Single click on the required file in the directory box (top right of the display)
- Press the 'Just play file' button.

1.11.7 Output file

After each analysis, an output file is generated called *output.dat*. It contains all the data necessary to generate the figure. It is a text file and can be read by any editor. It overwrites the existing *output.dat* file resulting from the previous analysis.

- The first column (headed 'Times (s)') is the list of times at which the output was sampled.
- The first line of text contains a list of the channel centre frequencies (Hz).
- The body of the matrix is the output from each channel arranged in columns.

2 The underlying model

The purpose of the model is to simulate human auditory function as closely as possible. After reading the signal from file, it is processed by HAP through three stages:

2.1 Outer/middle ear filter

The outer middle ear (pre-emphasis) filter is a set of four parallel IIR bandpass filters. The overall effect of the outer/middle ear was calculated by combining the outer ear transfer function as published in Moore et al. (1977) Fig. 2 with the middle ear function published as figure 3 of the same paper. The overall filter was published in Glasberg and Moore (2002).



Figure 4 Outer\ middle ear transfer function; comparison of psychophysical data with IIR filter used in the model.

2.2 Cochlear response

The **nonlinear** filterbank is the dual resonance nonlinear filterbank by Lopez-Poveda and Meddis (2001). The parameters used in the computations are those given in the paper.

The **linear** filterbank uses traditional gammatone filters. The width of the filters is set using psychophysical estimates (Moore and Glasberg, 1987).

The input to the filterbank is pressure (microPascals). The output is velocity of the basilar membrane (m/s).

The filter CFs are equally spaced on an ERB scale (see Moore et al, 1997).

2.3 Temporal integrator

The temporal integrator (in the form used here) is described by Oxenham and Moore (1994) and Oxenham and Plack (1998). More exactly, it is a weighted integration of the square of the velocity of the basilar membrane response(m/s). The software implements the integrator as a 3rd order low-pass filter with a cut-off frequency of 40 Hz.

The temporal integrator simulates forward masking effects.

The first 5 msec of the output is suppressed from the display because this reflects the start-up of the leaky integrator and FIR filters. This omission is intended purely to improve the appearance of the display.

The dB z-axis scale shown in the interface uses a reference value of $1e-12 (m/s)^2$ for the nonlinear filterbank and $1e-18 (m/s)^2$ for the linear filterbank. These values are arbitrary and chosen to show the output on a scale comparable to the input. Values below 0 dB are not plotted.

3 Calling HAP directly from MATLAB.

If the interface is not required, HAP can be called directly from a MATLAB program. The computations are carried out in the following function

[time, CFs, AMSoutput, errorMessage]=... HAP_MATLAB(speechFileName, params)

To use this function

- AMS must be installed because the MATLAB function runs the model using AMS.
- The folder *humanAuditoryPeriphery* must be on your MATLAB path. It is recommended that you make a copy of *humanAuditoryPeriphery* before using it.
- o humanAuditoryPeriphery must be the current directory

3.1 Input arguments

speechFileName the path of the *.wav* or *.aif* file. If the file is not in the current directory, give the full path name (e.g.' C:\...')

params a cell array of parameters, params.*. The current default values are: params.level=50; %peak level dB SPL params.minCF=100; %lowest centre frequency (Hz) params.maxCF=5000; %highest centre frequency (Hz) params.numCF=20; %number of channels params.modelType='nonLinear' %(alternatively,\linear\)

3.2 Output arguments

time	an array of times (s.) at which the signal was sampled
CFs	an array of the centre frequency values (Hz)
AMSoutput	a 2-D matrix (time/CF) of the output of the temporal integrator
errorMessage	normally an empty string. An error message is placed here if the function trapped an error. This message should be checked immediately after the execution of the routine.

3.3 Example program

function MATLABdemoHAP

% The name of the .wav or .aif file. % If the file is not in DSAM\AMS\HAP, give the full path name speechFileName='Roger.wav'

% params a cell array of parameters, params.*. The current default values are: params.level=50; % peak level dB SPL params.minCF=100; % lowest centre frequency (Hz) params.maxCF=5000; % highest centre frequency (Hz) params.numCF=20; % number of channels params.modelType= 'nonlinear' %(alternatively,'linear')

[time, CFs, AMSoutput, errorMessage]= HAP_MATLAB(speechFileName, params);

surf(time,CFs,log(AMSoutput))

4 Using HAP directly with AMS

You can run the AMS script directly without using either the GUI interface or MATLAB using the following sequence.

From the START menu launch AMS.

The AMS window should appear.

From the File pull down menu select Load parameter file (*.spf)

Then navigate to your copy of the *HAP* folder at C:\DSAM\AMS\HAP and select 'speechDisplayBM.spf'

📲 ams: speechDis (C:\allProgs\collaborator 💶 🗖 🗙							
<u>F</u> ile	<u>E</u> dit	⊻iew	<u>P</u> rogram	Help			
	GO						
				×			
				V			
				1.			

Click on the 'GO' button. You should see the following set of figures.



The first is the stimulus as seen by AMS. The second is the square of the velocity of the basilar membrane as computed using the DRNL method and the third is the output of the smoothed temporal window.

Individual parameters can be changed using the *edit/simulation parameters* pull-down menu. If you are not a regular user of AMS and you wish to explore it further, you can consult the tutorial materials supplied with the AMS installation in DSAM\AMS\tutorials..

Linear model.

You can run the linear model by using the alternative specification file.

- Navigate to the *humanPeriphery* folder and select 'speechDisplayLinear.spf'
- Click on the 'GO' button.

5 Problems??

Please report any problems to Ray Meddis (<u>rmeddis@essex.ac.uk</u>) after reading the following.

- The GUI interface will only work on MATLAB version 6 and later. It is not supported for earlier versions. To discover the version number of your MATLAB installation, type *version* in the command window
- The software was written and tested using the version of AMS supplied. It may not work with earlier versions of AMS.
- Did you put C:\DSAM\AMS\HAP on the MATLAB file pathe (using setpath)?
- The GUI expects to find the AMS executable (*ams_ng.exe*) at '*C:\DSAM\AMS*'. If you have put it somewhere else, you will need to change the line

```
amsdsam_path = 'c:\Progra~1\DSAM\AMS\';
```

currently at line 136 in the function *runAMS_HAP* at the bottom of the *matlabSpeechDemo.m* file

6 Bibliography

Glasberg, B. R., and Moore, B. C. J. (2002). "A model of loudness applicable to time-varying sounds," J. Audio Eng. Soc. 50, 331-342.

- Lopez-Poveda, E.A. and Meddis, R. (2001). 'A human nonlinear cochlear filterbank," Journal of the Acoustical Society of America, 110, 3107-3118.
- Moore, B. C. J. (1989). An Introduction to the Psychology of Hearing. London, Academic Press.
- Moore, B.C. J., Glasberg, B. R. and Baer, T. (1997). "A model for the prediction of Thresholds, Loudness, and Partial Loudness," Journal of the Audio Engineering Society, 45, 224-240.
- Moore, B.C.J. and Glasberg, B.R. (1987) "Formulae describing frequency selectivity in the perception of loudness, pitch and time," in Frequency Selectivity in Hearing, edited by B.C.J.Moore (Academic, London)
- Oxenham, A. J. and Moore, B.C.J. (1994). "Modelling the additivity of nonsimultaneous masking," Hearing Research, 80, 105-118.
- Oxenham, A. J. and Plack, C. J. (1998). "Basilar membrane nonlinearity and the growth of forward masking," Journal of the Acoustical Society of America, 103, 1598-1608.